

NAGY SZTOCHASZTIKUS MÁTRIXOK SZUBDOMINÁNS
SAJÁTÉRTÉKE¹

MOLNÁR GYÖRGY – SIMONOVITS ANDRÁS

MTA Közgazdaságtudományi Intézet

Bródy (1997) cikkét követve, egy olyan lineáris dinamikus rendszert vizsgálunk, amelynek mátrixa nagyméretű, nem-negatív és az egyes elemei eléggé hasonlóak egymáshoz. Bródy azt sejtette, hogy a rendszer konvergenciasebessége — a domináns és a szubdomináns sajátértékek hányadosának az abszolút értéke — gyorsan nő a szektorok számával. Ebben a cikkben sztochasztikus mátrixokra szorítkozva pontos eredményt bizonyítunk: ha a mátrix elemei legfeljebb $\tau/n^{1+\epsilon}$ -nel térnek el $1/n$ -től, akkor a szubdomináns sajátérték abszolút értéke legfeljebb τ/n' , ahol τ és ϵ tetszőleges pozitív állandók.

1. Bevezetés

A dinamikus vizsgálatok egyik legfontosabb kérdése a rendszer stabilitása. Nem elegendő azonban a rendszer stabilitását igazolni; azt is tudnunk kell, hogy a rendszer milyen gyorsan tart az egyensúlyi állapotához. (Mit érne egy közönséges mérleg, amely a helyes értéket több perc alatt állapítaná meg?) Ennek ellenére mind a hazai, mind a nemzetközi közgazdasági irodalomban meglehetősen keveset vizsgálták a különböző dinamikus rendszerek konvergencia-sebességét. Előzményként két cikket említünk meg: Atkinson (1969) és Simonovits (1978).

Nemrégiben Bródy (1997) a konvergencia-sebességgel kapcsolatban a következő kérdést tette föl: lehet-e valamilyen általános eredményt kapni, ha a gazdasági rendszer nagyon sok és egymáshoz hasonló egységből áll? A válasza egyszerű: ha a rendszer nagy és lineáris, valamint alkotórészei nagyon hasonlítanak egymásra, akkor a konvergencia nagyon gyors.

Tegyük föl, hogy dinamikus rendszerünk állapotát minden időszakban egy n -dimenziós vektor írja le, s az állapotok közti átmenetet egy $n \times n$ -es $A_n = A$ mátrix. (Ilyen rendszert vizsgált korábban Bródy (1992).) Ismert,

¹Köszönetet mondunk a korábbi változatok névtelen lektorainak, Simonovits Miklósnak és mindenképp Bródy Andrásnak, aki nemrég megjelent cikkének eredményeit a közlés előtt szóban elmondta és kéziratban megmutatta. Simonovits András köszönetét fejezi ki az OTKA T 019696 pályázat támogatásáért. Beérkezett: 1997. október 14.

hogy a relatív konvergencia-sebességet a mátrix abszolút értékben két legnagyobb sajátértékének — a domináns és a szubdomináns sajátértéknek — a hányadosa határozza meg. Ha az alkotórészek teljesen azonosak, akkor az A mátrix minden eleme azonos, normálás esetén $1/n$. Ezt a mátrixot $E_n = E$ -vel jelöljük.

Ebben a dolgozatban először bemutatjuk a sejtést, majd egy közeli problémával helyettesítjük a feladatot. Determinisztikus keretben már jogosan normalizálhatjuk az oszlopösszegeket, anélkül, hogy a függetlenséggel szembe kerülünk. Belátjuk, hogy az A_n mátrix szubdomináns sajátértékének az abszolút értéke legfeljebb akkora lehet, mint A_n és E_n ún. l_1 -távolsága. A tétel következménye azzal az esettel foglalkozik, amikor mindegyik elem közel esik a várható értékéhez, és ekkor igazolja a szubdomináns kicsinységét. Végül megmutatjuk, hogy a tétel éles.

A cikk felépítése a következő: a 2. pontban Bródynek a véletlen mátrixra vonatkozó sejtését ismertetjük, és e sejtésnek Wigner (1955), (1958), Füredi és Komlós (1981) cikkéhez való viszonyát vizsgáljuk. A 3. pontban kifejtiük a sztochasztikus mátrixokról szóló eredményeket, és utalunk a Diaconis és Stroock (1991) cikkkel való kapcsolatra.

2. Véletlen mátrixok

Legyen $A = A_n$ egy $n \times n$ -es mátrix, nem-negatív elemekkel: $A \geq 0$, $n > 1$. Föltesszük még, hogy a mátrix *primitív* (másképpen: aciklikus), azaz létezik olyan (1 és n közötti) p természetes szám, amelyre $A^p > 0$. Legyen μ_i az A mátrix nem-növekvő sorrendben vett i -edik sajátértéke: $|\mu_i| \geq |\mu_{i+1}|$, $i = 1, \dots, n-1$. A Perron–Frobenius tétel szerint létezik egyetlen egy domináns sajátérték, μ_1 , s ez pozitív: $|\mu_i| < |\mu_1|$, $i = 2, \dots, n$.

A konvergencia sebességét a második legnagyobb abszolút értékű, az ún. *szubdomináns* sajátérték és a domináns sajátérték aránya határozza meg.

Bródy (1997) azt vizsgálta, hogyan függ a szóban forgó hányados n -től! Szellemes fogással nem egy adott, hanem egy sereg *véletlen elemű* mátrixra vizsgálta a kérdést. Mint Bródy (1997) is hangsúlyozza, a véletlen mátrixok tipikus elemére általában élesebb eredményt kapunk, mint egy konkrét determinisztikus mátrixra.

Legyen $\{a_{i,j}\}$ ($1 \leq i, j \leq n$) korlátos, független és azonos eloszlású pozitív valószínűségi változók sorozata: $A_n = (a_{i,j})$ egy véletlen mátrix.

Bródy az elemzést a determinisztikus esettel, E_n -nel kezdte. Rámutatott, hogy ekkor A rangja 1, és az összes dominált sajátérték 0. Megállapította, hogy a Mises-iterációnál a perturbációs összetevők egyetlen egy iteráció alatt eltűnnek — függetlenül n értékétől.

A véletlen mátrixoknál bonyolultabb a helyzet. Persze megfelelő normálás után a nagy számok törvényéhez hasonlóan az elemek 1 valószínűséggel $1/n$ -hez "tartanak". A sajátértékek folytonosan függenek az elemektől, azonban a méretnövekedés miatt ez a tétel esetünkben nem alkalmazható. Bródy heurisztikus megfontolásokkal és számítógépes szimuláció segítségével mégis eljutott a következő

Sejtéshez. *A független elemekből álló, véletlen A_n mátrix szubdomináns sajátértékének és domináns sajátértékének hányadosa nullához tart, amint n tart a végtelenhez.*

Elben a cikkben Bródy kutatását folytatjuk. Történetünkben fontos szerepet játszik a szimmetrikus mátrixok kétparaméteres serege. Legyen σ és π két valós szám: $a_{ii} = \sigma$, ha $i = 1, \dots, n$ és $a_{ij} = \pi$, ha $i \neq j$. Jól ismert és könnyű igazolni, hogy A sajátértékei $\mu_1 = \sigma + (n-1)\pi$ és $\mu_2 = \sigma - \pi$, rendre 1 és $(n-1)$ -es multiplicitással. Eltekintve a valószínűtlen esetektől, feltehető, hogy 1 és $n-1$ független sajátvektor létezik, amely rendre μ_1 -hez és μ_2 -höz tartozik. A 3. pontban látni fogjuk, hogy ebben az esetben a sejtésnek megfelelően μ_2 tart a 0-hoz.

Egészen más okokból a fizikusok (Wigner (1955) és (1958)) már régóta vizsgálták a szimmetrikus nagy mátrixok spektrumát. Itt elegendő lesz a matematikus Füredi és Komlós (1981) éles eredményére utalni: legyen a főátló fölötti elemek szórása v^2 , és a várható érték mátrix tartozzék kétparaméteres családunkhoz. (A szimmetria miatt nem lehet minden elem független!) Ekkor

$$\frac{|\mu_2|}{|\mu_1|} \approx \frac{2v}{\sqrt{n}}.$$

Egyrészt Bródy sejtése általánosabb, mint Füredi és Komlós tétele, hiszen neincsak szimmetrikus várható értékű mátrixokra vonatkozik. Másrészt viszont speciálisabb, mert azonos eloszlású pozitív valószínűségi változók szerepelnek a sejtésben: $\sigma = \pi > 0$, míg Füredi és Komlós tetszőleges σ -t és π -t megenged, szabad előjellel.

Ismert, hogy a közgazdaságtanban a szimmetrikus mátrixok közel sem játszanak olyan fontos szerepet, mint a fizikában vagy a statisztikában. Nagyon gyakran föltehetjük viszont, hogy az átmeneti mátrix elemei pozitívak vagy nullák. Jelen cikkünkben éppen ezt a fontos esetet vizsgáljuk.

3. Sztochasztikus mátrixok

Mostantól kezdve csak nem-negatív elemű mátrixokat vizsgálunk, véletlen hatások nélkül, a következő normálással: mindegyik oszlop összege 1. Ezeket a

mátrixokat *sztochasztikusnak* nevezzük, és domináns sajátértékük egységnyi: $\mu_1 = 1$. A sztochasztikus mátrixok által származtatott dinamikát *Markov-láncnak* nevezik, és széleskörűen használják a valószínűségszámításban. Mint korábban említettük, minden primitív mátrixnak egyetlen domináns sajátértéke van.

Olyan mátrixokra fogjuk Bródy megfigyelését bizonyítani, amelyek közel esnek E -hez. (A közelséget a mátrix-normával fogalmazzuk meg.) Egy $n \times n$ -es U mátrix l_1 -normáját a következőképp definiáljuk: az elemek abszolút értékét véve $(|u_{ij}|)$, az új mátrix maximális oszlopösszege a norma, azaz $\|U\| = \max_j \sum_i |u_{ij}|$. Ennek megfelelően két mátrix, U és V l_1 -távolsága a különbség mátrix normája: $\|U - V\|$.

Kimondjuk a

Tétele. *A sztochasztikus A mátrix szubdomináns sajátértékének abszolút értéke legfeljebb akkora, mint az A és E mátrix l_1 -távolsága.*

Bizonyítás. Vegyük a $D = A - E$ különbségmátrixot, ahol $d_{ij} = a_{ij} - 1/n$. Mint ismert, az $\mathbf{1}$ ' összegző sorvektor minden sztochasztikus mátrix baloldali sajátvektora, tehát A -é is, E -é is. Az A mátrix dominált sajátvektorai, x_2, \dots, x_n mindegyike ortogonális $\mathbf{1}$ -re. Hasonlóan x_j az E mátrix 0 sajátértékhez tartozó $(n-1)$ -dimenziós sajátalterében fekszik: $E x_j = 0$. Ezért $\mu_j x_j = A x_j = E x_j + D x_j = D x_j$ maga után vonja, hogy μ_j D -nek is sajátértéke, $j = 2, \dots, n$. Ezen a ponton alkalmazhatjuk a következő elemi egyenlőtlenséget: egy négyzetes mátrix bármely sajátértékének az abszolút értéke legfeljebb akkora, mint a mátrix normája: $|\mu_2| \leq \|D\|$. Q.E.D.

Mivel $|d_{ij}| < \tau/n^{1+\epsilon}$ ($1 \leq i, j \leq n$) egyenlőtlenségekből következik $\|D\| < \tau/n^\epsilon$, egy egyszerűbb eredményt is megfogalmazhatunk.

Következmény. *Legyen τ és ϵ két valós szám, $\tau > 0$ és $\epsilon \geq 0$. Ha az $n \times n$ -es sztochasztikus A_n mátrix elemei kielégítik az $|a_{ij} - 1/n| \leq \tau/n^{1+\epsilon}$ ($1 \leq i, j \leq n$) egyenlőtlenségeket, akkor a szubdomináns sajátértékre teljesül $|\mu_2| \leq \tau/n^\epsilon$.*

Megjegyzések

1. A tétel és a következmény alap gondolata jól szemléltethető a kétparaméteres családon. Ahhoz, hogy sztochasztikus mátrixot kapjunk, egyparaméteres családra kell szorítkoznunk: $\sigma + (n-1)\pi = 1$. Ezért $\mu_1 = 1$ és $\mu_2 = 1 - n\pi$. További egyszerűsítésnél adódik E_n , ahol $\sigma = \pi = 1/n$ és $\mu_2 = 0$. Ha $\sigma = 1/n + \tau/n^{1+\epsilon}$, akkor $\mu_2 = \tau/n^\epsilon$.

2. Ha $\epsilon = 1$ és az elemek véletlen változók, akkor a Csebisev-egyenlőtlenség miatt a közelségi feltételünk megbízhatósági intervallumra egyszerűsödik, de az egyenlőtlenség csak valószínűleg teljesül. Ha $\epsilon = 0$ és $0 < \tau < 1$, akkor a közelségi feltétel pozitivitást is implikál.

3. A feltevés nem szükséges. Például legyen $\sigma = \beta/n$ és $\pi = (1 - \beta/n)/(n - 1)$. Ekkor $\mu_2 = (\beta - 1)/(n - 1)$. Itt az eltérések nagyságrendje $1/n$, de μ_2 nagyságrendje mégis $1/n$, nem pedig 1 (ami az $\epsilon = 0$ esetből adódna).

4. A 3. megjegyzés ellenére eredményünk más tekintetben éles. Legyen $n = 2m$, $a_{ij} = \pi$, ha $1 \leq i, j \leq m$ vagy $m + 1 \leq i, j \leq 2m$ és $a_{ij} = \sigma$ egyébként. Ahhoz, hogy sztochasztikus mátrixot kapjunk, fel kell tennünk, hogy $m(\pi + \sigma) = 1$. Ekkor elemi számolás után $\mu_2 = m(\pi - \sigma)$ adódik. A szimmetrikus $\pi = 1/n + \tau/n^{1+\epsilon}$ és $\sigma = 1/n - \tau/n^{1+\epsilon}$ esetben $\mu_2 = \tau/n^\epsilon$ áll. Éppen ez a tételben és következményében szereplő felső határ. Az érdekesség kedvéért megemlítjük, hogy az itt szereplő eltérés-mátrix 2-ciklikus (Rózsa, 1974), és az effajta mátrixok a Simonovits (1978) dolgozatban is kulcsszerepet játszottak.

5. Poincaré módszerét továbbfejlesztve, az elmúlt években több matematikus (például Diaconis és Stroock, 1991) érdekes felső korlátokat kapott a szimmetrikus (és közvetve, nem szimmetrikus) sztochasztikus mátrixok szubdomináns sajátértékére. Ezek az eredmények azonban olyan bonyolultak, hogy nem is tudjuk őket idézni. Csupán annyit említnünk meg, hogy az általános elméletben is az előzőleg említett 2-ciklikus mátrixtól való távolság a döntő. Talán jól érzékelteti a lényegét a kártyakeverés példája. Bár a kártyaeloszlások száma rendkívül nagy, néhány (7) keverési művelet után a kártyák jól elkeverednek. Ez arra utal, hogy a megfelelő sztochasztikus mátrix szubdomináns sajátértéke abszolút értékben nagyon kicsi.

Irodalom

1. Atkinson, A. B. (1969) "The Timescale of the Economic Models": How Long is the Long Run", *Review of Economic Studies* 36 137–152.
2. Bródy, A. (1992) "A pénzforgalom egy input-output modelje: kísérlet a multiplikátor-elmélet általánosítására és a multiplikátorok kiszámítására", *Közgazdasági Szemle* 39. 3. 197–207.
3. Bródy, A. (1997) "Leontief mátrixok legnagyobb sajátértékéről", *Sigma* 28, 1–6.
4. Diaconis, P. és Stroock, D. (1991) "Geometric Bounds for Eigenvalues of Markov Chains", *Annals of Applied Probability Theory* 1 36–61.
5. Füredi, Z. és Komlós, J. (1981) "The Eigenvalues of Random Symmetric Matrices", *Combinatorica* 1 233–241.
6. Rózsa, P. (1974) *Lineáris algebra és alkalmazásai*, Budapest, Műszaki Könyvkiadó.
7. Simonovits, A. (1978) "A decentralizált szabályozás maximális konvergencia-sebessége", *Sigma* 11 49–67.

8. Wigner, E. P. (1955) "Characteristic Vectors of Bordered Matrices with Infinite Dimensions", *Annals of Mathematics* 62 548–564.
9. Wigner, E. P. (1958) "On the Distribution of the Roots of Certain Symmetric Matrices", *Annals of Mathematics* 67 325–327.

THE SUBDOMINANT EIGENVALUE OF A LARGE STOCHASTIC MATRIX

Using intuition and computer experimentation, Bródy conjectured that the ratio of the subdominant eigenvalue to the dominant eigenvalue of a positive random matrix (with identically and independently distributed entries) converges to zero when the number of sectors tends to infinity. In this paper, we discuss the deterministic case and, among other things, prove the following version of this conjecture: if each entry of the matrix deviates from $1/n$ by at most $\tau/n^{1+\epsilon}$, then the modulus of the subdominant root is at most τ/n^ϵ , where τ and ϵ are arbitrary positive real parameters.

KÉT FUZZY OSZTÁLYOZÓ MÓDSZER¹

BORGULYA ISTVÁN

Janus Pannonius Tudományegyetem Közgazdaságtudományi Kar

Több kritériummal jellemzett alternatívák esetén az osztályozás, vagy klaszterek képzése gyakori feladat, több módszer létezik megoldásukra. Dolgozatomban két új heurisztikus módszert kívánok bemutatni, melyek több kritériummal jellemzett fuzzy, vagy éles (crisp) alternatívák osztályozását, valamint adott számú osztályra, klaszterre bontását teszik lehetővé.

Az osztályozó algoritmus minden alternatívához (elemhez) egy fuzzy rendező módszerrel, mint függvénnyel egy értéket rendel. Minden osztálynál értelmezi a centrumot, mint az osztály elemek függvényértékeinek átlaga, valamint minden osztályban egy prototípust, amely az osztály olyan eleme, melynek függvényértéke a legközelebb esik a centrum értékéhez. Az algoritmus a kritériumok súlyszámának változtatásával megkeresi az egyes osztályok prototípusait és centrumait. A klaszterképző algoritmus alap gondolata, lépései majdnem megegyeznek az osztályozó algoritmusával, de értelemszerűen a helyes osztályba sorolást nem tudja ellenőrizni. Mindkét rendszerrel a "tanulás" után további esetek is osztályozhatók

Más megközelítéssel készült módszerekkel, pl. két neurofuzzy osztályozó módszerrel összehasonlítva, a tesztelés hasonló eredményeket adott (97%-os pontosság). Az osztályozó algoritmus relatíve magas belső szórású osztályok elkülönítésére is alkalmas, osztályonként több csoport megkülönböztetésével.

1. Bevezetés

Több kritériummal jellemzett alternatívák (esetek) osztályozása, vagy klaszterek képzése az alapfeladatok közé tartozik. Több módszer létezik megoldásukra. Utóbbi időben különösen a fuzzy osztályozó, klaszterező algoritmusok terén olvashatunk újabb módszereket. A gyakori módszerektől eltérően, egy olyan módszert kívánok bemutatni, amely a neurális hálózatok tanulásához hasonló elemeket tartalmaz.

A neurális hálózatok tanulása vetette fel az ötletet, hogy az alternatívák rendezését fuzzy rendszerrel is meg lehet tanulni, és a több kritériummal

¹Készült az OTKA T18562 kutatás keretében. Beérkezett: 1998. január 24.

jellemzett alternatívák sorrendje, sorszámuk ismeretében, a kritériumok súlyszámának változtatásával, az ismert adatokból megtanulható [Borgulya 97b]. Ha az ismert osztályok azonosítóit az alternatívákhoz mint "sorszámot" rendeljük, az alternatívák osztályok alapján történő rendezése egyúttal osztályozást is fog jelenteni. Vagy ha ismeretlenek az osztályok, és k darab klaszterbe akarjuk sorolni az alternatívákat, a klaszterek centrumait "sorszámnak" tekintve, az alternatívák az egyes centrumok alapján szintén sorba rendezhetők.

Több kisebb példán ellenőrizve az elképzelést, az eredmények részben igazolták a feltevést. Egyszerű, könnyen szeparálható osztályok esetén olyan sorrendet kaptam, amely az azonos osztályba tartozó alternatívákat egy csoportba gyűjti, egymás után helyezve el őket. Komplikáltabb osztályozás esetén már a csoportok közt átfedések alakultak ki, fuzzy "csoportok" keletkeztek, ill. egyre több hibás osztályozás lépett fel. A precízebb osztályozás érdekében ezért az osztályba sorolásnál belső hasonlóságot is vizsgáltam, és a "sorszám" és a hasonlóság együtt határozta meg az alternatívák sorrendjét. Ez utóbbi esetben már lényegesen csökkent a hibák száma, az ellenőrző tesztek szerint 97-100%-os pontosságot sikerült elérni.

Mivel a területen számos algoritmus ismert, először tekintsük át nagy vonalakban a fuzzy klaszteranalízis és a fuzzy osztályozó algoritmusok csoportjait, majd részletesebben a két heurisztikus algoritmust és tesztelésének eredményeit.

2. Fuzzy klaszteranalízis és fuzzy osztályozás

A klaszteranalízis célja az adatok olyan csoportosítása, melyben a hasonló adatok egy osztályba (klaszterbe), a kevésbé hasonlóak különböző osztályokba kerülnek. Az elmúlt 20 évben számos klaszteranalízis eljárást (KAE) fogalmaztak meg. Az eljárások, melyek közt fuzzy módszerek is találhatóak, különbözőképpen csoportosíthatók. Egy lehetséges csoportosítás az alkalmazott technikák alapján [Höppner et al, 97], pl.:

- *Nem teljes KAE-ok.* Különböző geometriai, ábrázolási, projekciós technikák jellemzik. Az osztályok 2-3 dimenziós vetületeit állítják elő.
- *Determinisztikus KAE-ok.* Minden adat csak egy osztályba tartozik és az osztályok az adathalmazt teljesen felosztják.
- *Átfedő KAE-ok.* Minden adatot legalább egy osztályhoz hozzárendelnek, azaz egy adat több osztályhoz is hozzárendelhető.
- *Valószínűségi KAE-ok.* Egy valószínűségi eloszlás minden adatnál megadja, milyen valószínűséggel tartozik az adat valamely osztályhoz. Ha

a valószínűséget hozzátartozás foknak értelmezzük, akkor ezek az eljárások fuzzy KAE-nak is tekinthetők (a valószínűségek összege egy).

- *Lehetségességi KAE-ok.* Minden adatnál megadják, hogy milyen lehetőségességi, vagy hozzátartozási fokkal tartozik valamely osztályhoz.
- *Hierarchikus KAE-ok.* Az eljárások több lépésben egyre finomabb bontását állítják elő az induló osztálynak, ill. fordítva: az induló kis osztályokat egyre nagyobb osztályokká olvasztják össze.
- *Objektív függvényt alkalmazó KAE-ok.* Az objektív függvény olyan cél, vagy értékelő függvény, melynek az optimumát keressük. Az objektív függvény minden osztály felosztáshoz egy hibaértéket, vagy jóság mértéket rendel, ami alapján a legkedvezőbb osztályozást választhatjuk ki.

A fuzzy KAE-ok többsége az objektív függvényt alkalmazó rendszerekhez tartozik, és az objektív függvény alkalmazása mellett általában valószínűség-eloszlás, vagy lehetőségességi fokok alapján történik a klaszterek kialakítása.

A fuzzy KAE-ok többsége akár valószínűségi, akár lehetőségességi klaszter felosztással definiálható. Az első, és legismertebb algoritmus a *Fuzzy-c Means-algoritmus* (FCM). A módszer éles adatokon megfogalmazott első változatát Duda és Hart (1973) fogalmazta meg, a jelenleg használatos változatát 1992-ben publikálták [Bezdek et al, 92]. A FCM eljárás kúpalakú pontthalmazokat képes elkülöníteni a p -dimenziós térben. Minden klasztert egy középponttal jellemez, melynek koordináta pontjai a "prototípusok". A klasztereket a prototípusok képviselik, és az osztályba sorolás az adatok és a prototípusok távolsága alapján történik. Az eljárás feladata egy iterációs eljárás során az optimális klaszter középpontok meghatározása a hozzá tartozó hozzátartozási fokokkal együtt. Az iteráció mindaddig folytatódik, amíg a középpontok stabilá nem válnak.

A FCM-ből az alkalmazott távolságdefiníció megváltoztatásával a kúpformájú klaszterek helyett ellipszis, ill. ellipszoid alakú klaszterek felismerése is elérhető. E módosítást a Gustafson-Kessel algoritmus fogalmazza meg [Gustafson et al, 79]. A FCM algoritmus, ill. a Gustafson-Kessel algoritmus további, olyan változatairól is beszámolhatunk, melyek már egy fuzzy kovariancia mátrixot használnak fel a klaszterek alakjának, nagyságának beállítására.

A FCM algoritmus egyes változatai, különösen a képfeldolgozásban használhatók fel sikeresen. A *lineáris klaszter eljárások* (pl. Fuzzy-c-Variations-algoritmus, vagy adaptive-Fuzzy Clustering-algoritmus) a képeken az egyeneseket, mint klasztereket ismerik fel. A *shell-klaszter eljárások* (pl. fuzzy-C-Shell-algoritmus, vagy fuzzy-C-Ellipsoidal-Shell-algoritmus stb.), még tovább

lépnek. Körök, ellipszisek, parabolák, hiperbolák kontúrjait, mint klasztereket képesek felismerni. A FCM továbbfejlesztett változatai speciális törtvonalak kontúrjait, mint klasztereket ismernek fel (pl. Fuzzy-C-Rectangular-Shell-algoritmus, vagy Fuzzy-C-2-Rectangular-Shell-algoritmus). Így téglalapok, szabályos sokszögek, vagy téglalapok metszeteként előállítható törtvonalakat, mint klaszterek kontúrjait ismerik fel.

E fuzzy KAE-ok mind iteratív eljárások, melyek az iteráció minden lépésben finomítják a hozzátartozási fokok és a klaszter középpontok becslését. A módszerek egy másik csoportja egy lépésben, iteráció nélkül számolja ki az előbbi értékeket. Ilyen módszer pl. Yager és Filev Mountain Clustering algoritmus, amely ismeretlen számú klaszter meghatározására készült [Yager et al, 94].

Az osztályozás nagyon hasonló probléma a klaszteranalízishez. Egyes feladatoknál a klaszter algoritmussal kapott osztályok egybeesnek a várt osztályokkal, azaz a kétféle feladat eredménye ugyanaz. Összetettebb feladatoknál azonban a klaszter algoritmusok nem adnak kielégítő eredményt az osztályozási feladatra. Kombinálni kell a klaszteranalízis módszereit más módszerekkel, vagy teljesen más nézőpontból kell megközelíteni a problémát.

Az osztályozást megvalósító fuzzy rendszerek (FR) az esetek többségében szabályalapú rendszerek, melyek, minden osztályhoz legalább egy szabállyal rendelik az elemeket. A probléma formálisan a következőképpen írható le:

Legyenek $X = (x_1, x_2, \dots, x_n)$ n dimenziós elemek, amelyek a $C = C_1, C_2, \dots, C_k$ osztályokba tartoznak (Bármely két osztály metszete üres). Az elemek koordinátáit írjuk le az A_1, A_2, \dots, A_n fuzzy halmazok segítségével, és az egyes osztályokat kijelölő R_1, R_2, \dots, R_k szabályok formája legyen

$$R_i: \text{IF } (A_1 \text{ is } x_1) \text{ AND } (A_2 \text{ is } x_2) \text{ AND } \dots \text{ AND } (A_n \text{ is } x_n) \text{ THEN class is } i,$$

ahol a class változó a konklúzióként kapott C_i osztály sorszámát adja. Minden szabály előállít valamilyen osztálysorszámot, és az osztályozáshoz a legjobban illeszkedő osztályt kell kiválasztani. Azt a szabályt kell tehát keresni, amelynek premisszáit a legjobban kielégíti az adott elem, azaz amelyik osztályhoz a legnagyobb hozzátartozási fokkal besorolható. Ha az A_1, A_2, \dots, A_n fuzzy halmazok tartalmazási függvényeit $\mu_1, \mu_2, \dots, \mu_n$ -nel jelöljük, akkor az i -edik szabálynál a hozzátartozási fok

$$\mu_{R_i}(x_1, x_2, \dots, x_n) = \min\{\mu_1(x_1), \mu_2(x_2), \dots, \mu_n(x_n)\}.$$

A keresett osztályt a legnagyobb $\mu_{R_i}(x_1, x_2, \dots, x_n)$ értékű osztály adja.

A szabályok meghatározása többféleképpen történhet. Ha 2 dimenzióban, lineárisan szeparálható osztályokról van szó, akkor az osztályokat elválasztó egyenest egy lépcsősfüggvénnyel közelíthetjük, és minden "lépcsőhöz" egy-egy szabályt írhatunk fel. Összetettebb, több osztályt tartalmazó n -dimenziós osztályozási feladatoknál azonban ez az út nem járható.

Az újabb osztályozási módszerek neurofuzzy, vagy KAE-ok segítségével osztályoznak. A neurofuzzy rendszerek az osztályozásra kerülő adatokból tanulják meg a szükséges fuzzy halmazokat és szabályokat. A klaszteranalízisen alapuló módszerek a felismert klaszterek alapján generálnak halmazokat és szabályokat. Neurofuzzy osztályozó módszert alkalmaz pl. a NEFCLASS rendszer, amely egy 3 rétegű fuzzy perceptron segítségével állítja elő a fuzzy osztályozó rendszer halmazait, szabályait [Nauck et al, 95].

A felsorolt fuzzy osztályozó rendszerek fuzzy halmazokat és szabályokat generálnak az adatok alapján. Az ismertetésre kerülő súlyszám tanulással történő osztályozásnál mások a kiinduló feltételek. Itt ismertek az elemek leírásához szükséges kritériumok éles adatai, vagy fuzzy halmazai, és centrumokat, prototípusokat, súlyszámokat kell keresni.

A heurisztikus fuzzy osztályozó algoritmus leginkább a FCM algoritmushoz hasonló. Az osztályozáshoz osztály centrumokat, prototípusokat határoz meg. Az osztályok elemeit elsősorban hasonlóság alapján állapítja meg, mely egy lehetőségességi klaszter felosztásnak felel meg. Értékelő függvényként olyan hibafüggvényt alkalmaz, amely egy fuzzy rendező módszer eredményei alapján ellenőrzi a sorrendet és a centrumok stabilitását. Eltérően a FCM-től, a centrumok, prototípusok keresésére a kritérium súlyok változtatását is felhasználja. E ponton a neurofuzzy rendszerekhez hasonló az algoritmus.

3. Fuzzy osztályozás súlyszám tanulással

Az osztályozó algoritmus alapgondolata

Rendeljünk minden alternatívához (elemhez) egy fuzzy rendező módszerrel, mint függvényvel egy értéket, mely alapján az alternatívák sorrendje ellenőrizhető. Az osztályok elemeit hasonlóság alapján soroljuk egy-több csoportba, és rendeljük minden csoporthoz egy centrumot és egy prototípust. A centrum az osztály (ill. csoportbeli) elemek függvényértékeinek átlaga, a prototípus pedig az osztály olyan eleme, melynek függvényértéke a legközelebb esik a centrum értékéhez. Az algoritmus a kritériumok súlyszámának változtatásával megkeresi, megtanulja az egyes osztályokat, csoportokat jellemző prototípusokat, centrumokat. **Hogy az algoritmus a relatíve magas belső szórású osztályokat is fel tudja ismerni, osztályonként a hibás osztályozástól függően automatikusan több csoport definiálását is lehetővé teszi.**

Osztályozás esetén minden elemnek ismerjük az osztályát. A fuzzy rendszernek (FR) ezt a helyes osztályt kell a kritériumok értékei alapján felismernie és az elemhez rendelnie. Mivel az osztályok a centrumokkal és a kapcsolódó prototípusokkal egyértelműen jellemezhetőek, olyan algoritmust

kell megfogalmazni, amely egy elemhez azt az osztályt rendeli, melynek prototípusához a leghasonlóbb és ez egyúttal azonos a helyes osztállyal is.

Az osztályozást fokozatosan tanulja meg rendszer. Egyenként, véletlenszerűen veszi elő az elemeket, és minden elemet hasonlósági mérték alapján valamelyik osztályba (csoportba) sorol. Hogy a centrumok egy stabil érték felé konvergáljanak, a kritériumok súlyszámait úgy módosítja, hogy az elem új függvényértéke a megfelelő centrum felé közeledjék. A súlyszámokat minden olyan esetben módosítja, ha az osztályba sorolás téves, vagy az utolsó két elemhez tartozó függvényértékek sorrendje eltérő osztály centrumaik sorrendjétől (a súlyszám módosítás a neurális hálózatok delta szabályához hasonló képlettel történik). Az algoritmus mindaddig folytatódik, amíg adott pontossággal nem ismeri fel az osztályokat, és a centrumok stabillá nem válnak (változásuk értéke adott küszöbszám alá nem csökken).

A hibás sorrend megállapításához figyelembe kell venni, hogy

- az egy osztályba tartozó elemek sorrendje közömbös, bármilyen sorrendjük elfogadható;
- az osztályok azonosító sorszám, és az osztályok rendezésnél kapott sorrendje eltérő lehet;
- az egyes osztályok, ill. az osztályt alkotó csoportok sorrendjét centrumaik véletlenszerűen kialakult sorrendjével azonosíthatjuk.

Nézzük e heurisztikus algoritmust közelebbről. Jelöljük az osztályozásra kerülő elemeket a_1, a_2, \dots, a_n -nel, és ismert osztályait pedig jelöljük az $1, 2, \dots, k$ sorszámokkal. Az elemeket leíró kritériumokat jelöljük K_1, K_2, \dots, K_m -mel, és az l -edik kritériumot, mint nyelvi változót értelmezzük. A választott fuzzy rendező módszer (FRM) a K_1, K_2, \dots, K_m nyelvi változók alapján minden a_i elemhez egy y_i valós számot rendel ($i = 1, 2, \dots, n$).

Az algoritmus megfogalmazásához három lényeges pontban kell dönteni: a hasonlóság kezelésében, a súlymódosítás szabályában és az alkalmazható FRM választásában.

Hasonlóság

A hasonlóságot azonos formában kezelhetjük éles és fuzzy elemek esetén. Éles kritériumokkal adott a_i, a_j elemeknél a hasonlóság mértéke legyen

$$H(a_i, a_j) = 1/(1 + d(a_i, a_j)), \quad (1)$$

ahol $d(a_i, a_j)$ az éles elemek euklideszi távolsága. Fuzzy kritériumok esetén a hasonlóság mértékére ismét az (1) összefüggés alkalmazható, de a távolság-képletben Munda (1995) szemantikus távolságát használjuk fel. E szerint két

tetszőleges a_i, a_j elem távolsága ($p = 2$ esetén az „euklideszi” távolság):

$$d(a_i, a_j) = |a_i - a_j| = \left[\sum_{l=1}^N w_l * \left(\iint |x - y| f_l(x) g_l(y) dy dx \right)^\mu \right]^{\frac{1}{\mu}}$$

ahol folytonos, konvex tartalmazási függvényeket feltételezve, esetünkben $f_l(x) = c_{1l} * \mu_{K_l}(x)$ és $g_l(x) = c_{2l} * \mu_{K_l}(x)$ (c_{1l}, c_{2l} valós számok), és f_l, g_l -re teljesülni kell:

$$\int f_l(x) dx = \int g_l(x) dx = 1.$$

Az algoritmus egy elem osztályba tartozását hasonlóság alapján fogja megállapítani. Amelyik prototípushoz a leghasonlóbb, annak az osztályához fog az eset tartozni. Tekintettel arra, hogy a relatíve magas belső szórású osztályoknál egy prototípussal nem lehet minden elemet jól jellemezni, osztályonként több prototípust is alkalmazhatunk. Ekkor hibás osztályozás esetén újabb prototípust vezethetünk be.

Súlymódosítás

Az osztály centrumok alapján való rendezést az algoritmus véletlenszerűen kiválasztott elemek alapján ellenőrzi. Ha az utoljára vizsgált két elem sorrendje nem megfelelő, "kis lépésekkel" a helyes sorrend irányába változtatja az elemekhez rendelt y számokat. Ehhez minden olyan kritérium súlyát változtatja, és a helyes irányban növeli, ill. a másik irányban csökkenti, ahol a két elemnél az azonos kritériumok értékei eltérőek.

A hasonlóság alapján történő osztály hozzárendeléseknél sorrendhiba két-féle okból keletkezhet:

1. két tetszőleges, különböző osztályhoz tartozó elemnél az osztályok centrumainak sorrendje ellentétes az elemekhez az FRM által rendelt függvényértékek sorrendjével, vagy
2. egy elemnél az algoritmus által meghatározott osztály nem azonos a helyes osztállyal.

Ha az i és j -edik elemnél az osztályozás helyes, csak a FRM által hozzárendelt függvényértékek sorrendje hibás, az éles, vagy fuzzy kritériumoknak megfelelően módosítani kell a súlyokat. A súlyszám változtatás mértékét:

- az utolsó két elemnél az azonos kritériumok értékeinek különbsége,
- az utolsó két elemhez tartozó osztályok centrumainak különbsége,

- az utolsó elem hasonlósága a helyes prototípusához: $H(a_t, p_z)$,
- a "tanulási ráta", amely megadja a súlyszám változtatás egységét

befolyásolja. Ha a vizsgált párnál az a_i és a_j elemek sorrendje nem megfelelő, akár több kritérium súlyszámát is szükséges módosítani. Ha az algoritmus által meghatározott osztály eltér a helyes osztálytól, korrigáljuk úgy a hibát, hogy az elemet hasonlóbbá tesszük a helyes osztály prototípusához. A súlymódosító szabály ebben az esetben is ugyanaz, csak előző elemek a helyes osztály prototípusát kell választani.

A súlymódosítás szabálya különbözik éles, ill. fuzzy kritériumok esetén. Éles kritériumnál a súlymódosítás képlete:

$$\Delta w_l = -\eta * \text{sign}(y_i - y_j) * (k_{li} - k_{lj}) * (oc - c_z)H(a_i, p_z), \quad (2)$$

ahol η a tanulási ráta, k_{li} , k_{lj} az előző és az utolsó kritériumérték az l -edik kritériumnál, p_z a helyes osztály prototípusa, és oc , c_z az elemek helyes centrumait jelöli.

Fuzzy kritériumok esetén nem tudunk a k_{li} , k_{lj} valós értékek alapján következtetni a szükséges súlymódosítás irányára, nagyságára. A kritériumok, mint fuzzy halmazok közti kisebb, nagyobb relációt viszont meghatározhatjuk minden kritérium esetén Munda (1995) javaslata alapján az

$$\text{int}_l = \iint (x - y)f_l(x)g_l(y) dy dx$$

integrál előjele alapján. Ha pozitív, akkor a nagyobb, különben a kisebb reláció igaz a kritériumok aktuális értékei közt. A súlymódosításnál ezt felhasználva a (2)-es képlet új alakja:

$$\Delta w_l = -\eta * \text{sign}(y_i - y_j) * \text{int}_l * (oc - c_z)H(a_i, p_z), \quad (3)$$

Megjegyzés: Az algoritmus programozásánál egyszerűsítve számoltam: a szemantikus távolságot közelítésként folytonos helyett diszkrét esetre alkalmaztam, és int_l helyett a

$$\mu_K(x) * \mu_K(y) * (x - y)$$

képlettel számoltam, ahol x , y a kritérium input értékei.

A rendező módszer

FRM-nek az algoritmus tesztelése közben a max-min módszert [Yager 78], valamint az "osztályozó módszert" [Borgulya 95, 97a] választottam. Éles kritériumok esetén a két módszer hasonló eredményt adott. Tekintettel arra, hogy az osztályozó módszer fuzzy kritériumoknál általánosabban használható, az osztályozó módszert alkalmaztam végül az algoritmusnál. Nézzük a továbbiakban e rendező módszert vázlatosan.

Olyan több kritériummal jellemzett a_1, a_2, \dots, a_n alternatívákat kell rendezni, ahol a K_1, K_2, \dots, K_m kritériumok súlyozottak és a kritérium értékek fuzzy halmazok (g_1, g_2, \dots, g_m a kritériumokhoz tartozó súlyszámok, és a súlyszámok maximális értéke 1).

E probléma megoldását kétféleképp közelítjük meg:

- speciális esetként feltesszük, hogy a kritériumok értékei az oktatásban alkalmazott osztályzatokkal kerülnek megadásra, vagy ilyen osztályzatokká transzformálhatók;
- általános esetben magukat a kritérium értékeket tekintjük osztályzatnak anélkül, hogy az oktatásban alkalmazott osztályzatok tulajdonságaival rendelkezzenek.

Mindkét esetben egyetlen számot, "extra osztályzatot" rendel a módszer minden alternatívához, melyek ez alapján rendezhetők.

Általános eset

Legyen $X = \{a_1, a_2, \dots, a_n\}$ az alternatíváknak egy véges halmaza, legyen $K = \{K_1, K_2, \dots, K_m\}$ a fuzzy kritériumoknak egy véges halmaza, legyenek továbbá g_1, g_2, \dots, g_m a kritériumokhoz tartozó súlyszámok, a súlyszámok maximális értékét egynek választva.

Tekintsünk X felett minden K_j fuzzy kritériumot nyelvi változónak ($1 \leq j \leq m$) és legyen

$$K_j = \{S_{j1}, S_{j2}, \dots, S_{jP_j}\},$$

ahol $S_{j1}, S_{j2}, \dots, S_{jP_j}$ a nyelvi változó értékei. Az a_i alternatívát ($1 \leq i \leq n$) a K_j kritérium $S_{j1}, S_{j2}, \dots, S_{jP_j}$ fuzzy halmazzaival értékelhetjük.

Definiáljuk továbbá a E eredményhalmazt a kritérium halmazok uniójaként:

$$E = \bigcup_{j=1}^m K_j.$$

Definiáljuk a következő FAM szabályokat:

(g_1^2)	IF $k_1 = S_{11}$	THEN $E = S_{11}$
(g_1^2)	IF $k_1 = S_{12}$	THEN $E = S_{12}$

(g_1^2)	IF $k_1 = S_{1p_1}$	THEN $E = S_{1p_1}$
(g_2^2)	IF $k_2 = S_{21}$	THEN $E = S_{21}$
(g_2^2)	IF $k_2 = S_{22}$	THEN $E = S_{22}$

(g_2^2)	IF $k_2 = S_{2p_2}$	THEN $E = S_{2p_2}$

(g_m^2)	IF $k_m = S_{m1}$	THEN $E = S_{m1}$
(g_m^2)	IF $k_m = S_{m2}$	THEN $E = S_{m2}$

(g_m^2)	IF $k_m = S_{mp_m}$	THEN $E = S_{mp_m}$

ahol (g_i^2) $i = 1, 2, \dots, m$ a szabályok súlyszámai, és k_1, k_2, \dots, k_m egy alternatíva adatai.

Az előállított fuzzy rendszer a szabályok leképezéseit aggregálva (T -konorma), a súlypont defuzzifikáló eljárással minden a_i alternatívához egy y_i értéket rendel, mely alapján rendezhetők az alternatívák [Borgulya 97a].

Speciális eset

Az általános osztályozó módszer speciális eseteként a rendezni kívánt alternatíváknál a kritériumok értékei legyenek egységesen, nem szükségképpen azonos fokozatú, fuzzy halmazként értelmezett osztályzatok [Borgulya 95]. Az osztályzatok (S_1, S_2, \dots, S_p) legyenek szimmetrikus háromszögekkel, vagy Gauss-függvényekkel megadott fuzzy halmazok a [jegy-1, jegy+1] intervallum felett, ahol jegy=1, 2, \dots , p lehet. (Az E halmaz most csak a lehetséges p darab S_1, S_2, \dots, S_p érték uniójából áll). A speciális esetben az alternatívákhoz rendelt y_i értékek osztályzatként értelmezhetők, és rendezhetők segítségükkel az alternatívák.

A heurisztikus algoritmusban fuzzy kritériumok esetén az osztályozó módszer általános, éles kritériumok esetén a speciális változatát alkalmazzuk. Éles kritériumok esetén a kritérium értékeket transzformáljuk egységesen, pl. a $[0, 5]$ intervallumra, és utána alkalmazható a FRM.

A heurisztikus fuzzy osztályozó algoritmus lépései

Legyen az osztályok száma k , és engedjük meg $k + t$ számú prototípus alkalmazását (t lehet nulla is). Legyen r_0 egy adott hasonlósági érték, amely

egy osztály több csoportra bontását szabályozza, legyen ϵ a centrumok stabilitásának ellenőrzésénél a küszöbszám, és legyen γ a még elfogadható hibás osztályozások száma. Jelöljük az elemek helyes osztályait ho_1, ho_2, \dots, ho_n -nel.

1. Definiáljuk az osztályok (ill. csoportok) induló üres halmazait $C = \{C_i\}$, $i = 1, 2, \dots, k+t$. Legyenek c_1, c_2, \dots, c_{k+t} , valamint p_1, p_2, \dots, p_{k+t} az osztályok centrumai, valamint prototípusai. A C_i halmazhoz tartozó centrum, prototípus, osztály sorszám legyen c_i, p_i, o_i . Legyen $c_i = 0, p_i = 0$, és $o_i = 0$ ($i = 1, 2, \dots, k+t$).
2. Válasszunk véletlenszerűen egy w_1, w_2, \dots, w_m súlyszám sorozatot. Válasszunk véletlenszerűen két indexet: i, j -t. Legyen az első a_i elem nulla. Az induló pár (a_i, a_j) . Legyen $oc = 0$ és $y_i = 0$.
3. Legyen FRM: $a_j \rightarrow y_j$
4. Hasonlóság vizsgálat. Ha $H(a_j, p_z) = \max_q H(a_j, p_q)$ (alkalmazva az (1) képletet éles, vagy fuzzy esetben), ahol $q, z \in [1, k+t]$, akkor az a_j -hez rendelt osztály o_z . Legyen FRM: $p_z \rightarrow y$.
5. Prototípus módosítás.
 - a) Ha $(ho_j = o_z) \wedge (c_z - y_j < c_z - y) \wedge (H(a_j, p_z) \geq ro)$ akkor $p_z = a_j$, és ha $a_j = p_x$ ($x = 1, 2, \dots, z-1, z+1, \dots, k+t$), akkor $c_x = 0, o_x = 0$.
 - b) Ha $(ho_j \neq o_z) \wedge (c_l = 0) \wedge l = ho_j$ akkor $p_l = a_j, c_l = y_j, o_l = ho_j$ és $z = l$. Különb, ha $(ho_j \neq o_z) \wedge (\exists c_l)(c_l = 0) \wedge l = k+1, \dots, t$, akkor $p_l = a_j, c_l = y_j, o_l = ho_j$ és $z = l$.
6. Súlymódosítás.
 - a) Ha $(ho_j = o_z) \wedge (oc < c_z \wedge y_i > y_j) \vee (oc > c_z \wedge y_i < y_j)$, akkor éles kritériumok esetén a (2), fuzzy kritériumok esetén a (3) súlymódosító képletet alkalmazzuk.
 - b) Ha $(ho_j \neq o_z) \wedge (\exists x)(o_x = ho_j)$ akkor $p_x \rightarrow a_i, y_x = c_x, oc = 2c_x, z = x$, és éles kritériumok esetén a (2), fuzzy kritériumok esetén a (3) súlymódosító képletet alkalmazzuk.
7. Súlyok normalizálása.

$$\|W\| = \sum_{i=1}^m w_i \quad \text{és} \quad w_i = \frac{w_i}{\|W\|}, \quad i = 1, 2, \dots, m.$$

8. $i = j$, $oc = c_z$ és válasszunk véletlenszerűen újabb a_j elemet. Az új pár ismét (a_i, a_j) .
9. Ellenőrizzük az eredményt az iteráció minden k -dik lépésében. Számoljuk újra a centrumok értékeit: $uc_1, uc_2, \dots, uc_{k+t}$,

$$uc_l = \frac{1}{db_l} \sum_{s=1}^{db_l} y_s,$$

ahol db_l az l -edik osztály (csoport) elemeinek száma, és FRM: $a_s \rightarrow y_s$ az osztály elemeihez rendelt értékek. Ha a

$$\left(\sum_{l=1}^{k+t} |c_l - uc_l| < \epsilon \right) \wedge (\text{hiba} < \gamma)$$

hibafüggvény értéke igaz (ahol "hiba" a hibás osztály-meghatározások száma), vége az eljárásnak. Különben $c_l = uc_l$, $l = 1, 2, \dots, k + t$ és folytatás a 3. pontnál.

Ha véletlenszerűen vesszük elő az elemeket, megfelelő ro , η , ϵ és γ választása esetén az algoritmus konvergens lesz, és minden osztály centruma külön-külön konvergál egy stabil értékhez. E ponton hasonló a rendszer működése a neurális háló tanulásához. Pl. a tanulási ráta értékétől függően gyorsabb, lassabb lehet a konvergencia, vagy esetleg nem lesz konvergens a tanulási folyamat. A különböző paraméter értékek megválasztása függ az adott példától. Általában a $ro = 0.95$, $\eta = 0.0005$, $\eta = 0.01$ és $\gamma = n/10$ értékek mellett konvergens a tanulási folyamat.

A rendszer alkalmazása

Ha a tanulási folyamat befejeződik, megjeleníthetők az eredmények, ill. újabb esetek osztályozhatók a kész rendszerrel. Az algoritmus a centrumok, prototípusok, súlyszámok ismertetében minden alternatívánál megállapítja melyik osztály prototípusához a leghasonlóbb, és a prototípusához tartozó osztályt rendeli az alternatívához. Ez a hozzárendelés tartalmazhat hibákat. Enyhítve a hasonlóság mértékét, és pl. minden alternatívánál a legnagyobb hasonlósági érték -0.05 értékkel számolva, a nagyobb hasonlósági intervallum alapján egyes alternatíváknál két, vagy több osztályt is megadhat a rendszer mint lehetséges osztályt. Természetesen a hasonlóság tetszőlegesen tovább enyhíthető. (Egy osztály alternatíváihoz rendelt hasonlósági értékek olyan függvényt képeznek, amely egy osztály tartalmazási függvényéhez hasonlóan értelmezhető).

4. Fuzzy klaszter algoritmus

A fuzzy osztályozó algoritmus természetes kiterjesztéseként adódik a lehetőség, hogy osztály kialakításra, klaszterképzésre is alkalmassá tegyük az algoritmust. Klaszter kialakítás esetén a fuzzy rendszernek k darab különböző osztályba, klaszterbe kell sorolnia az alternatívákat úgy, hogy a leghasonlóbbak egy osztályba kerüljenek.

Ha a fuzzy osztályozó algoritmushoz hasonlóan itt is egy alternatívához azt az osztályt rendeljük, melynek prototípusához a leghasonlóbb, az előző feladatra visszavezethető a probléma. Ellentétben az osztályozással, az osztály hozzárendelés helyességét nem tudjuk ellenőrizni. Ellenőrizhető viszont most is bármely két elemhez rendelt függvényértékének sorrendje a hozzájuk rendelt osztályok centrumai alapján, valamint az egyes alternatívák és a prototípusok hasonlósága. Ha a pillanatnyi sorrend hibás, alkalmazható az előző súlymódosító szabály éles, vagy fuzzy változata.

Ha véletlenszerűen vesszük elő az elemeket, ez a heurisztikus algoritmus is konvergens lesz és folyamatosan módosítja az egyes osztályok elemeit, prototípusait, centrumait mindaddig, amíg a centrumok stabillá nem válnak.

Az algoritmus lépései

1. Definiáljuk az osztályok (ill. csoportok) induló üres halmazait $C = \{C_i\}$, $i = 1, 2, \dots, k$. Legyenek c_1, c_2, \dots, c_k , valamint p_1, p_2, \dots, p_k az osztályok centrumai, valamint prototípusai. A C_i halmazhoz tartozó centrum és prototípus legyen c_i, p_i . Legyen $c_i = 0, p_i = 0$ ($i = 1, 2, \dots, k$).
2. Válasszunk véletlenszerűen egy w_1, w_2, \dots, w_m súlyszám sorozatot. Válasszunk véletlenszerűen két indexet: i, j -t. Legyen az első a_i elem nulla. Az induló pár (a_i, a_j) .
3. Legyen FRM: $a_j \rightarrow y_j$
4. Hasonlóság vizsgálat. Ha $H(a_j, p_z) = \max_q H(a_j, p_q)$ (alkalmazva az (1) képletet éles, vagy fuzzy esetben), ahol $q, z \in [1, k]$, akkor legyen $a_j \in C_z$. Legyen FRM: $p_z \rightarrow y$.
5. Prototípus módosítás.
Ha $(c_z \neq 0) \wedge (c_z - y_j < c_z - y)$ akkor $p_z = a_j$, és ha $a_j = p_x$ ($x = 1, 2, \dots, z - 1, z + 1, \dots, k$), akkor $c_x = 0$. Különben, ha $(\exists z)(c_z = 0)$, akkor $p_z = a_j$, és $c_z = y_j$.

6. Súlymódosítás.

a) Ha $(oc < c_z \wedge y_i > y_j) \vee (oc > c_z \wedge y_i < y_j)$, akkor éles kritériumoknál a (2), fuzzy esetben a (3) súlymódosító képletet alkalmazzuk.

7. Súlyok normalizálása.

$$\|W\| = \sum_{i=1}^m w_i \quad \text{és} \quad w_i = \frac{w_i}{\|W\|}, \quad i = 1, 2, \dots, m.$$

8. $i = j$, $oc = c_z$ és válasszunk véletlenszerűen újabb a_j elemet. Az új pár ismét (a_i, a_j) .

9. Ellenőrizzük az eredményt az iteráció minden k -dik lépésében. Számoljuk újra a centrumok értékeit: uc_1, uc_2, \dots, uc_k ,

$$uc_l = \frac{1}{db_l} \sum_{s=1}^{dl_l} y_s,$$

ahol db_l az l -edik osztály elemeinek száma, és FRM: $a_s \rightarrow y_s$ az osztály elemeihez rendelt értékek. Ha a

$$\sum_{l=1}^k |c_l - uc_l| < \epsilon$$

hibafüggvény értéke igaz, vége az eljárásnak. Különben $c_l = uc_l$, $l = 1, 2, \dots, k$, és folytatás a 3. pontnál.

Megjegyzés: Felvethető az algoritmus olyan továbbfejlesztése, amely osztályonként több prototípust alkalmaz. Ennek előnyei csak további teszteléssel mutathatók ki.

A rendszer alkalmazása

Ha a tanulási folyamat befejeződik, a kész rendszerben az eredmények az osztályozó algoritmushoz hasonlóan jeleníthetők meg, ill. újabb alternatívák osztályozhatók. Az algoritmus minden alternatívánál megállapítja melyik osztály prototípusához a leghasonlóbb az alternatíva és a prototípushoz tartozó osztályt rendeli hozzá. Enyhítve a hasonlóság mértékét, nagyobb hasonlósági intervallum alapján már az egyes alternatívákhoz két, vagy több osztályt is megadhat a rendszer, mint lehetséges osztályt. (Egy osztály alternatíváihoz rendelt hasonlósági értékek most is olyan függvényt képeznek, amely egy osztály tartalmazási függvényéhez hasonlóan értelmezhető).

5. Mintapéldák, tesztelés

Fuzzy osztályozó algoritmus alkalmazása

Első példa. Összehasonlításuképpen a többiek által benchmark-ként alkalmazott IRIS adatsort választottam teszt adatsornak, és eredményemet a NEFCLASS, valamint FuNeGen I. neuro-fuzzy osztályozó programokkal [Nauck 1995, Halgamuge 1994] is összevettem (A programok szabadon elérhetők az interneten).

A 150 elemű IRIS adatsor, melyet Fischer (1934) publikált először, az írisz virág 3 típusát, osztályát írja le 4-4 adat alapján. E feladat összetettebb osztályozási problémát jelent, mivel két osztálya lineárisan nem szeparálható. Az adatsor egy részlete:

5.1	3.5	1.4	0.2
4.9	3.0	1.4	0.2
4.7	3.2	1.3	0.2
4.6	3.1	1.5	0.2

ahol az első k_1 kritérium a $[4, 8]$, k_2 a $[2, 5]$, k_3 az $[1, 7]$ és k_4 a $[0, 3]$ intervallumban veszi fel értékeit. Mivel a példához kapcsolódó fuzzy halmazokra nincs semmilyen megkötés, külön-külön a $[0, 5]$ intervallumra transzformáltam az adatokat és minden kritériumnál a nyelvi változók értékeit, mint "osztályzatokat" az $[i-1, i+1]$ ($i = 1, 2, \dots, 5$) intarvallumok felett szimmetrikus háromszög alakú tartalmazási függvényekkel adtam meg. Fuzzy rendező módszernek az "osztályozó"-módszert választottam.

Az adatsor két egyenlő részből: IRIS1 és IRIS2 áll (IRIS1: 75 elem, melyben 25 elem tartozik minden osztályhoz, és IRIS2: 75 elem, melyben szintén 25 elem tartozik minden osztályhoz). Training-re az adatsor első felét választottam és az adatsor második felével teszteltem a kész fuzzy rendszert. A súlymódosításon alapuló osztályozó módszer (jelöljük FCW-vel) szinte azonos pontosságú eredményeket adott a másik két módszerrel (1. ábra):

	IRIS1	IRIS2
FuNeGen I.	99.0%	99.0%
NEFCLASS	96.0%	97.3%
FCW	97.3%	97.3%

1. ábra IRIS adatsor összehasonlító eredményei 3 prototípus esetén (a százalékok a helyes osztályozásokat jelentik).

Ha enyhítjük a hasonlóság mértékét és minden alternatívánál a kapott maximális hasonlósági értéknél (MHi) 0.05-el kisebb értékeket is elfogadunk,

a hibás osztályozások száma háromról kettőre csökken és a korábbi tévesen osztályozott alternatívához két osztályt fog rendelni. Ha a hasonlóságot tovább enyhítjük, csak a $[MHi-0.3, MHi]$ intervallumba eső értékeknél tűnnek el a téves osztályozások, viszont ekkor az alternatívák 49%-ánál (37-nél) 2, vagy 3 osztályt rendel már minden alternatívához. A hasonlóság mértékétől függően tehát a hibás osztályozások, ill. egy alternatívához rendelt osztályok száma változó (2. ábra):

	Hasonlóság mértéke $> MHi - \delta$				
	$\delta = 0$	$\delta = 0.05$	$\delta = 0.1$	$\delta = 0.2$	$\delta = 0.3$
téves osztályozás	3	2	2	2	0
több osztály hozzárendelés	0	3	9	23	37

2. ábra Az eredmények és a hasonlóság kapcsolata ($i = 1, 2, \dots, 75$.)

Az eljárás konvergenciáját a 3. ábra szemlélteti, amely különböző paraméter értékek esetén mutatja a konvergencia gyorsaságát (iterációk száma) és az osztályozások helyességét (százalékok). A +++ jelölés az adott paraméterek mellett nem konvergens tanulást mutatja (ebben az esetben egyébként a pontosság kb. 75%-os).

t. ráta	ro	Prototípus szám	ϵ	Kritérium	IRIS1	IRIS2	Iteráció
0.0005	-	3	0.001	Fuzzy	97.3%	97.3%	6751
0.05	0.85	5	0.001	Fuzzy	96.0%	96.0%	2326
0.05	0.9	5	0.001	Fuzzy	80.0%	80.0%	885
0.005	0.85	5	0.001	Éles	98.5%	98.5%	5116
0.0005	0.9	5	0.1	Éles	98.5%	98.5%	976
0.0005	0.95	11	0.1	Fuzzy	98.5%	100.0%	724
0.0005	0.9	5	0.01	Fuzzy			+++

3. ábra Paraméterek és a konvergencia összefüggése

Második példaként állattartási peresetek osztályozását nézzük. L. Philipps (1994) számol be arról, hogy állattartással kapcsolatos bírói döntéseket vizsgálva, az állattartó érdekei és felelőssége alapján hat jellemző tulajdonsággal, kritériummal írhatók le az esetek. A hat tulajdonság (k_1, k_2, \dots, k_6) értékeit fuzzy értékeknek tekintve, fuzzy logika műveletekkel állította elő a bírói döntést. 18 eset adatait felhasználva csupán 3 esetben tért el az eredmény a bírói döntéstől.

Ezen adatok (4. ábra) osztályozhatók is, bár lényegesen összetettebb osztályozási problémát jelentenek, mint az IRIS adatok osztályozása. Fuzzy osztályozó módszerekkel a bírói igen/nem döntésnek megfelelően pontosan besorolhatók két osztályba az esetek.

Tekintsük most is az adatokat osztályzatoknak, és alkalmazzuk az előbbi példa fuzzy halmazait. A kritérium értékekre a következő megfeleltetéseket fogadjuk el: null =0; nagyon csekély=0.5; csekély=1; közepes = 2.5; magas = 3.75; nagyon magas =4.5; teljes = 5. Osztályozva az eseteket mind a FCW, mind a FuNeGen I. és a NEFCLASS osztályozó programokkal, az eredmények jobbakk lettek, mint a fuzzy logika műveletekkel kapott értékek. A bírói döntést nulla, vagy egy hibával reprodukálták (5. ábra).

	k1	k2	k3	k4	k5	k6	bírói d.
1.	n.csekély	csekély	csekély	null	teljes	közepes	nem
2.	teljes	null	csekély	magas	null	magas	igen
3.	csekély	közepes	közepes	közepes	teljes	közepes.	igen
4.	teljes	csekély	teljes	teljes	null	magas	igen
5.	null	n.magas	közepes	magas	n.magas	n.magas	igen
6.	teljes	n.csekély	közepes	csekély	n.csekély	n.csekély	igen
7.	null	null	közepes	csekély	n.csekély	közepes	nem
8.	közepes	magas	csekély	magas	teljes	közepes	igen
9.	csekély	közepes	teljes	null	teljes	null	nem
10.	teljes	teljes	magas	magas	null	teljes	igen
11.	csekély	közepes	null	n.csekély	teljes	n.csekély	nem
12.	teljes	teljes	teljes	teljes	teljes	teljes	igen
13.	null	közepes	csekély	teljes	teljes	null	igen
14.	teljes	közepes	magas	null	null	teljes	igen
15.	null	közepes	n.csekély	magas	csekély	teljes	igen
16.	null	csekély	közepes	n.csekély	közepes	csekély	nem
17.	csekély	magas	n.csekély	közepes	teljes	közepes	igen
18.	teljes	teljes	teljes	teljes	közepes	teljes	igen

4. ábra. Jogesetek adatai

	Helyes osztályozások	
	száma	százaléka
FuNeGen I. (10 generált szabállyal)	17	94.4%
NEFCLASS (18 generált szabállyal)	18	100%
FCW	18	100%

5. ábra. Jogesetek osztályozása

(A hasonlóság mértékét e feladatnál nem kell enyhíteni, minden osztály besorolás helyes). A 100%-os pontosságot 0.0005 tanulási ráta, 2 prototípus, fuzzy kritériumok alkalmazásával, $\gamma = 1$ esetén 1218 iteráció után kaptuk.

Harmadik példaként nézzünk egy olyan feladatot, amely csak plusz prototípusok alkalmazásával oldható meg elfogadható pontossággal. Válasszuk a relatíve magas belső szórású osztályokat tartalmazó következő feladatot: két éles kritériummal adott elemeket kell két osztályba sorolni, ahol az elemek: $C_1 = \{(0, 6), (12, 0), (1, 8), (1, 10), (14, 4)\}$; $C_2 = \{(3, 12), (15, 6), (16, 8), (5, 16), (17, 10)\}$

Az osztályozó algoritmus a prototípusok számától függően különböző gyorsasággal, pontossággal oldotta meg a feladatot. A 100%-os megoldást már 4 prototípus alkalmazása mellett nyújtotta (6. ábra)

t. ráta	ro	Prototípus szám	Pontosság	Iteráció
0.005	-	2	80%	2400
0.0005	0.6	4	100%	55
0.0005	0.9	5	90%	310
0.05	0.8	5	80%	1160
0.0005	0.8	6	100%	55
0.0005	0.95	9	90%	2060

6. ábra Különböző paraméterértékeknél kapott eredmények ($\epsilon = 0.01$, $\gamma = 2$).

Fuzzy klaszter algoritmus alkalmazása

Osztály kialakítására nézzük először ismét az *IRIS adatsort*. Ha az előző megközelítést választjuk: azaz a $[0, 5]$ intervallumra transzformálását a kritérium értékeknek, akkor alkalmazhatjuk az osztályképző algoritmust. Mivel az *IRIS* adatsor osztályozását ismerjük, utólag ellenőrizhetjük az osztályok megfelelő kialakítását.

Ismét az „osztályozó” fuzzy rendező módszert választva az algoritmus mind a training-halmaznál (*IRIS1*), mind a teszhalmaznál (*IRIS2*) azonos, 95%-os pontossággal az ismert osztályozást adta vissza. A hasonlóság mértékének enyhítésével, $[MHi - 0.05, Mhi]$ intervallumbeli hasonlósági értékeknél már eltűntek a hibák, és 7 alternatívát 2-2 osztályba sorolt (összehasonlításra más fuzzy klaszterező szoftverek nem álltak rendelkezésre).

Osztály kialakításra *második példaként* nézzünk egy szokványos, valós számokkal adott feladatot (éles változat). Legyen adott 25 ország néhány statisztikai adata (7. ábra), és soroljuk 5 osztályba az országokat.

	k1	k2	k3	k4	k5	k6	k7
Afganisztán	40.4	18.7	21.6	1881.6	41.0	42.0	7.6
Bangla-desh	42.2	15.5	26.7	119.0	56.9	55.9	5.53
Kambodzsza	41.1	16.6	24.8	130.0	47.0	49.9	4.71
Kína	21.2	6.7	14.5	32.0	68.0	70.9	2.45
Ciprus	188.6	8.2	10.4	12.0	73.9	78.3	2.33
Egy. Arab E.	22.8	3.8	19.0	26.0	68.0	72.9	4.82
Hong-Kong	11.7	4.9	6.8	6.1	68.6	80.0	1.23
India	30.5	10.2	20.3	91.0	52.5	52.1	4.2
Indonézia	28.6	9.4	19.2	75.0	58.5	62.0	3.48
Irán	42.5	11.5	31.0	108.1	55.75	55.04	5.2
Irak	42.6	7.8	34.8	69.0	63.0	64.8	6.35
Izrael	22.3	6.3	16.0	9.7	73.87	77.44	3.03
Japán	9.9	6.7	3.3	4.5	75.9	81.77	1.54
Kuvait	26.8	2.2	24.6	15.6	71.2	75.4	4.03
Mongólia	36.1	8.8	27.3	68.0	60.0	62.5	5.0
Nepál	39.6	14.8	24.8	128	50.88	48.1	5.94
Pakisztán	30.3	8.1	22.2	107.7	59.04	59.2	6.48
Fülöp-sz.	33.2	7.7	25.5	45.0	62.5	66.1	4.33
Szaud-A.	42.1	7.6	34.5	71.0	61.7	65.2	7.17
Szingapúr	17.8	5.2	12.6	7.5	68.7	74.0	1.97
Sri-L.	21.2	6.2	125.1	19.4	67.78	71.6	2.96
SzírIA	44.6	7.0	37.6	48.0	64.42	68.05	6.76
Thaiföld	22.3	7.7	15.3	28.0	63.82	68.05	2.6
Töröko.	29.2	8.4	20.8	76.0	62.5	65.8	3.69
Vietnam	31.8	9.5	22.3	64.0	63.66	67.89	

7. ábra. Országok adatai (ahol k1: élveszületési ráta, k2: halálozási ráta, k3: természetes növekedési ráta, k4: csecsemőhalandósági ráta, k5: férfiak várható élettartama, k6: nők várható élettartama, k7: termékenységi ráta)

Bár most az adatokat nem tekintjük fuzzy értékeknek, az algoritmusnál felhasznált fuzzy rendező módszerhez továbbra is szükséges a fuzzy értelmezés. Akárcsak az előző esetekben, alkalmazzuk most is a fuzzy "osztályzatokat" és transzformáljuk a kritériumok értékét a $[0, 5]$ intervallumra. Az osztály kialakító algoritmus éles távolságokkal fog számolni, de a függvényértékeket továbbra is a FRM határozza meg.

Az eredmény ellenőrzéshez két különböző módszerrel is készítettem osztályokat. A statisztika egyik hierarchikus klaszteranalízis programjával, valamint egy neurális háló módszerrel, Kohonen önszervező térképével (SOM). Mind a három eredmény hasonló lett, de a fuzzy kluszterek tértek el a legjobb-

ban a többitől. A fuzzy kluszter algoritmus eredménye egyébként legjobban a statisztikai módszer eredményéhez hasonlítható (8. ábra).

Ez az eredmény már nem olyan jó, mint az osztályozás eredményei. A fuzzy algoritmussal csak 75-80%-ban kapunk azonos eredményt az éles módszerek eredményével. Ha átfedő osztályokat alkalmazunk, és a hasonlóság mértékét enyhítjük, a [MHi-0.15, Mhi] intervallumbeli értékekre már 12 országot két, esetleg három osztályba sorol és a kritikus, korábban eltérést mutató helyeken "megfelelő" osztályt is választhatunk. (Az algoritmus most is hasonló konvergenciát mutatott a fuzzy osztályozó algoritmusával: 0.0005 tanulási ráta és $\eta = 0.001$ mellett 1024 iteráció után stabilizálódtak a centrumok.)

klaszteranalízis	neurális háló	fuzzy klaszter
AAABC	AAABC	AAACC
BCDDA	BCADA	CCDDA
DCCBD	DCCCD	BCCCD
AAEDC	AAEDC	ADEBC
BEBDD	CEBDE	CECDD

8. ábra. Osztályozás eredményei

5. Összefoglalás

A bemutatott két heurisztikus algoritmus közös jellemzője, hogy a rendezni kívánt alternatívákból a neurális hálók tanulásához hasonlóan határozza meg az osztályozás, vagy klaszterezés alapját képező információkat. Az osztályozás helyességét olyan hibafüggvénnyel ellenőrzi, amely egy fuzzy rendező módszer eredményeit is felhasználja. A helyes osztályozás kialakítása érdekében a kritériumok súlyait, melyek általában a fuzzy rendező módszer szabályinak súlyai is egyben, a neurális hálózatok delta szabályához hasonló képlettel változtat egy iteratív folyamatban.

A tanulási folyamat végén minden osztályhoz egy centrumot, az osztályra jellemző prototípust, ill. a fuzzy rendező módszer szabályainak súlyait határozzák meg az algoritmusok. A fuzzy osztályozó algoritmus egy osztályon belül több csoportot és csoportonként külön centrumot, prototípust is értelmezhet; ezzel a nagyobb belső szórású osztályok helyes felismerésére is képes. A tanulás konvergenciája a neurális hálózatokhoz hasonló tulajdonságú.

Egy kész rendszer az éles, vagy fuzzy kritériumok ismeretében, valamint a kapott prototípusok, súlyok felhasználásával a továbbiakban önállóan képes osztályozni az elemeket. A tesztek eredményei alapján a fuzzy osztályozást,

klaszterképzést 97-100%-os pontossággal valósítják meg, lehetővé téve átfedő, fuzzy osztályok értelmezését is. Ezen eredmények pl. a neurofuzzy osztályozó módszerekkel hasonló pontosságúak.

Irodalom

1. Bezdek, J. C., Hathaway, R. H.: Numerical Convergence and Interpretation of the Fuzzy C-shells Clustering Algorithm. *IEEE Trans. Neural Networks* 3 1992. 787–793.
2. Borgulya, I.: Egy osztályozaton alapuló fuzzy rendező módszer. *Szigma XXVI*, 1995. pp. 117–132.
3. Borgulya, I.: A Ranking Method for Multiple Criteria Decision Making. *International Journal of Systems Science*, 1997a. 28. pp. 905–912.
4. Borgulya, I.: Generalisation Weightnumbers of Criteria by Fuzzy Systems. In: *Proceeding of the 19th Int. Conf. Information Technology Interfaces 1997b*. Pula. pp. 99–104.
5. Duda, R., Hart, P.: *Pattern Classification and Scene Analysis*. Wiley, New York, 1973.
6. Fischer, R. A., The Use of Multiple Measurements in Taxonomic Problems. *Annual Eugenics*, 7. 1936. (Part II), pp. 179–188.
7. Gustafson, E. E., Kessel, W. C.: Fuzzy Clustering with a Fuzzy Covariance Matrix. *IEEE CDC, San Diego, kalifornien* 1979. 761–766.
8. Halgamuge, S. K., Glesner, M., *Neural Networks in Designed Fuzzy Systems for Real World Applications*. *Fuzzy Sets and Systems*, 65. 1994. pp.1–12.
9. Höppner, F., Klawonn, F., Kruse, R.: *Fuzzy-Clusteranalyse* Vieweg, Braunschweig/Wiesbaden, 1997.
10. Munda, G., *Multicriteria Evaluation in a Fuzzy Environment*. Heidelberg, Physica-Verlag, 1995
11. Nauck, D., Kruse, R., NEFCLASS - A Neuro-Fuzzy Approach for the Classification of Data. In: *Proc. of the 1995 ACM Symposium on Applied Computing*. ACM Press, pp. 461–465.
12. Philipps L.: Kompensatorische Verknüpfungen in der Rechtsanwendung – ein Fall für Fuzzy Logic. In: *Martinek M., Schmidt J., Walde E.: Festschrift für Günther Jahr zum siebenzigsten Geburtstag*. J. C. B. Mohr, Tübingen 1994. pp. 169–180.
13. Yager, R. R., Fuzzy Decision Making Including Unequal Objectives. *Fuzzy Sets and Systems*. 1, 1978. pp. 87–95.
14. Yager R. R., Fliev D.: Generation of Fuzzy Rules by Mountain Clustering. *Journal of Intelligent & Fuzzy systems*, Vol. 2, No. 3, pp. 209–219.

TWO FUZZY CLASSIFICATION METHODS

Two heuristic fuzzy algorithms are shown in this paper: a fuzzy classification and a fuzzy cluster algorithm. The algorithms classify multiple-criteria fuzzy or crisp alternatives and the task of the classification is traced back to ranking of alternatives, as well as to learning weight numbers of criteria similarly to neural networks. The algorithm is suitable to separate classes with relatively high intrinsic scatter.

INSTABILITÁS ÉS KAOTIKUS VISELKEDÉS MOLNÁR ÉS SZIDAROVSKY DISZKRÉT DINAMIKUS FOGYASZTÓI-TERMELŐI MODELLJÉBEN¹

BACSI ZSUZSANNA

Pannon Agrártudományi Egyetem

A jelen dolgozat a Szidarovszky és Molnár (1994) által definiált diszkrét dinamikus termelői-fogyasztói modell viselkedését vizsgálja az instabil tartományban, adaptív és extrapolatív termelői várakozások esetén, feltételezve, hogy a piaci ár a piaci kereslet kínálat egyensúlytalanságának mértékével arányosan változik. A modellt kiegészítettük a piaci ár nemnegativitását biztosító alsó korláttal. Az így lineárisból szakaszonként lineárisra tett, de még mindig viszonylag egyszerű modellszerkezet rendkívül változatos viselkedést tesz lehetővé, többek között kaotikus viselkedés előállítására is képes, és a piaci egyensúlytalanságra reagáló K paraméter értékére igen érzékenyen reagál, ami ezen paraméter megválasztásának fontosságát hangsúlyozza.

1. Bevezetés

Determinisztikus káosz alatt mindig olyan jelenséget, illetve folyamatot értünk, ami sztochasztikus hatást nem tartalmaz, de hosszabb távon nem jelezhető előre. A kaotikus jelenségek ugyanakkor többnyire bizonyos határok között maradnak. E kettőből következik, hogy a kaotikus jelenség az adott határok között viszont szélsőségesen változhat (*Benhabib, 1992*). Ez egy piac esetében azt jelenti, hogy a termék ára szabálytalanul ingadozhat, ami a termelő szemszögéből erősen veszteséges és erősen nyereséges időszakokat jelent. Az előrejelezhetetlen, szeszélyes árak növelik a termelők kockázatát, és hosszú távon gyakran jelentős költségnövekedést okoznak. A kaotikus viselkedés a gazdasági modellezésben nem ritka jelenség, azonban előfordulása csak nemlineáris modellekben tapasztalható. Megmutatjuk, hogy jelen esetben egy lineáris modellben a vizsgált jelenségre természetesen adódó nemnegativitási feltétel már elegendő nemlinearitást visz a modellbe a kaotikus viselkedés előfordulásához.

¹Beérkezett: 1998. március 20.

2. A matematikai modell

A Molnár és Szidarovszky (1994) által kidolgozott egytermékes piacmodellt vettük vizsgálat alá. A modell lehetőséget biztosít több termelő szerepeltetésére a piacon, és figyelembe veszi ezeknek az aktuális piaci árra vonatkozó várakozásait. A modellnek azon változatával dolgoztunk, amely a piac egyensúlytalanságát tételezi fel. (A piaci egyensúlyt feltételező modellváltozat kaotikus viselkedés szempontjából való elemzését ld. Vizvári et al, 1997.).

A tényleges piaci ár alakulásának vizsgálata során feltettük, hogy a piaci ár kizárólag nemnegatív lehet, azaz a zéró érték egy alsó korlát. Lehetséges lenne egy felső korlát bevezetése is a piaci árra, azonban a nulla alsó korlát alkalmazása lényegében ugyanolyan jellegű eredményekhez vezet.

Jelölések:

N	a termelők száma a piacon
$p_{k,t}^c$	a k termelő által a t időpontra becsült ár
α_k	rögzített paraméter, amit k termelő a becsült ár meghatározásához használ ($0 \leq \alpha_k$)
$p_{\min} = 0$	az ár alsó korlátja, azaz az alsó intervenció ár
p_{\max}	a felső intervenció ár (a jelen vizsgálatokban nem használjuk)
$x_{k,t}$	a k termelő által termelt mennyiség a t időpontban
p_t	a piaci ár a t időpontban
d_t	a t időpontbeli teljes keresett mennyiség
$K > 0$	a piaci egyensúlytalanságra való reagálás mértéke
D, d	a keresleti függvény paraméterei
$C_{k,t}$	a k termelő költségfüggvénye a t időpontban
B_k, b_k, c_k	a k termelő költségfüggvényének paraméterei

A költségfüggvény kvadratikus, tehát

$$C_{k,t} = B_k x_{k,t}^2 + b_k x_{k,t} + c_k$$

ahol $B_k > 0$, mert ez biztosítja, hogy egy bizonyos mennyiség felett a költség már a lineárisnál gyorsabban nő. Mivel minden termelő a hasznát kívánja maximalizálni, ezért

$$x_{k,t} = \operatorname{argmax}\{p_{k,t}^c x_{k,t} - C_{k,t}\} \quad k = 1, \dots, N,$$

azaz

$$x_{k,t} = \frac{p_{k,t}^c - b_k}{2B_k} \quad k = 1, \dots, N.$$

Tehát látható, hogy a piacra vitt mennyiség a becslült árral lineárisan nő. Feltételezzük, hogy a fogyasztók ismerik a mindenkori piaci árat, és a keresleti függvény lineáris, azaz valamely $D < 0$ mellett

$$d_t = p_t D + d.$$

A piaci ár a piaci egyensúlytalanság mértékével arányosan változik, azaz

$$y_t = p_t D + d - \sum_{k=1}^N x_{kt}.$$

és

$$p_t = p_{t-1} + K y_{t-1}.$$

A termelők a piaci ár becslésére kétféle módszert alkalmaznak. Az *adaptív* modell szerint:

$$\forall k : 0 \leq \alpha_k \leq 1$$

esetén

$$\forall k : p'_{k,t} = \alpha_k p_{t-1} + (1 - \alpha_k) p'_{k,t-1}.$$

Egy másik lehetőség az ú.n. *extrapolatív* modell alkalmazása:

$$\forall k : 0 \leq \alpha_k$$

esetén

$$\forall k : p'_{k,t} = \alpha_k p_{t-1} + (1 - \alpha_k) p_{t-2}.$$

Itt tehát α_k értéke nagyobb is lehet, mint 1. Ha $\alpha_k = 2$, akkor $p'_{k,t} = p_{t-1} + (p_{t-1} - p_{t-2})$, vagyis a módszer az utolsó árváltozást extrapolálja a következő évre is, innen a neve.

3. A modell illesztése a magyar burgonyapiacra

A számítások alapjául a magyar burgonyapiac esetét vizsgáltuk. Mindösszesen két termelő szerepel, az egyik a nagytermelők, a másik a kistermelők csoportját reprezentálja. Ezek költségfüggvényeinek paramétereit az alábbiak szerint becsültük meg: a nagytermelő esetén: $B_1 = 0.014$, $b_1 = 3200$, $c_1 = 20\,000\,000$, a kistermelő esetén: $B_2 = 0.0058$, $b_2 = 4000$, $c_2 = 20\,000$. Végül $\alpha_1 = 0.6$, $\alpha_2 = 0.5$.

Statisztikai adatokra támaszkodva a d értéke 2 millió tonnának vehető. Ezzel szemben D erősen változik, annyi állítható biztonságosan, hogy értéke a

$[-100, 0]$ intervallumba esik (mértékegysége t^2/Ft). A paraméterek becslésére vonatkozóan ld. *Vizvári et al. (1997)*.

Molnár és Szidarovszky (1994) megadja mind az adaptív, mind az extrapolatív modell esetében a stabilitás feltételeit, amennyiben az α_k paraméterértékek azonosak minden termelő esetében. *Bessenyei (1996)* a modell olyan változatát elemzi, amelyben a piacon 4 termelő van jelen, azonos költségfüggvénnyel, de különböző árbecslési módszerrel (azaz teljes információs, naív, adaptív, illetve extrapolatív becsléssel).

A jelen dolgozat a modell egy olyan speciális esetét vizsgálja, amelyben a piacon két termelő van jelen, eltérő költségfüggvénnyel, azonos árbecslési módszerrel, de eltérő súlyokkal. Ezenkívül a piaci árra explicit alsó korlátként bevezettük a zéró értéket, azaz feltettük, hogy a piaci ár sohasem lehet negatív. A fenti módon meghatározott modellváltozat viselkedését vizsgáljuk, különös tekintettel az instabil viselkedés előfordulásának lehetőségére, és az esetleges kaotikus viselkedés előfordulására. Megmutatjuk, hogy az alsó korlát bevezetése jelentősen megváltoztatja a modell viselkedésének jellegét, és amellet, hogy csakis a valóságban is értelmezhető nemnegatív értékeket eredményez, a korlátozás nélküli modellváltozatnál jóval gazdagabb dinamikával rendelkezik.

3.1 Az adaptív modell esete

Két termelő esetén az áralakulást a következő egyenletrendszerrel írhatjuk le:

$$p_{t+1} = p_t(1 + KD) - \frac{K}{2B_1}p_{1,t}^c - \frac{K}{2B_2}p_{2,t}^c + K(d + \frac{b_1}{2B_1} + \frac{b_2}{2B_2}) \quad (1)$$

$$p_{k,t+1}^c = \alpha_k p_{k,t} + (1 - \alpha_k)p_{k,t}^c \quad k = 1, 2 \quad (2)$$

A rendszer mátrixa és sajátérték-egyenlete felírható hasonlóan a *Molnár és Szidarovszky* által közöltékhez, így némi számolás után a következő egyenlet adódik:

$$\lambda^3 + a_1\lambda^2 + a_2\lambda + a_3 = 0 \quad (3)$$

ahol

$$a_1 = \alpha_1 + \alpha_2 - 3 - KD \quad (4)$$

$$a_2 = 3 - 2(\alpha_1 + \alpha_2) + \alpha_1\alpha_2 + \frac{K}{2}(\frac{\alpha_1}{2B_1} + \frac{\alpha_2}{2B_2}) - KD(\alpha_1 + \alpha_2 - 2) \quad (5)$$

$$a_3 = (KD + 1)(\alpha_1 + \alpha_2 - 1 - \alpha_1\alpha_2) + \frac{K\alpha_1\alpha_2}{2}(\frac{1}{B_1} + \frac{1}{B_2}) - \frac{K}{2}(\frac{\alpha_1}{B_1} + \frac{\alpha_2}{B_2}) \quad (6)$$

Molnár és Szidarovszky említett dolgozatában közli a rendszer stabilitásának szükséges és elégséges feltételeit, eszerint:

Lemma. Az (1) és (2) által meghatározott rendszer pontosan akkor stabil, ha (3) karakterisztikus egyenlet együtthatóira vonatkozó alábbi (7)-(11) egyenlőtlenségek egyszerre teljesülnek:

$$1 + a_1 + a_2 + a_3 > 0 \quad (7)$$

$$1 - a_1 + a_2 - a_3 > 0 \quad (8)$$

$$3 + a_1 - a_2 - 3a_3 > 0 \quad (9)$$

$$3 - a_1 - a_2 + 3a_3 > 0 \quad (10)$$

$$1 - a_3^2 + a_1a_3 - a_2 > 0 \quad (11)$$

Némi számolás után a (7) egyenlőtlenség a következő ekvivalens alakra hozható:

$$K\alpha_1\alpha_2\left(\frac{1}{2B_1} + \frac{1}{2B_2} - D\right) > 0$$

amiről látható, hogy $D < 0$ és $K > 0$ miatt mindig teljesül, azaz nem jelent korlátozást.

A (8) egyenlőtlenség a következő alakban írható:

$$\frac{2\alpha_1\alpha_2 - 4\alpha_1 - 4\alpha_2 + 8}{K} + D\alpha_1\alpha_2 - \frac{\alpha_1\alpha_2}{2}\left(\frac{1}{B_1} + \frac{1}{B_2}\right) > 0$$

A (9) egyenlőtlenség a következő alakban írható:

$$\frac{2\alpha_1\alpha_2}{K} + D(3\alpha_1\alpha_2 - 2\alpha_1 - 2\alpha_2) + \frac{\alpha_1}{B_1} + \frac{\alpha_2}{B_2} - \frac{3\alpha_1\alpha_2}{2}\left(\frac{1}{B_1} + \frac{1}{B_2}\right) > 0$$

ami az adott α_k és B_k paraméterértékek mellett minden $K > 0$, $D < 0$ mellett teljesül. A (10) egyenlőtlenség a következő alakban írható:

$$\frac{\alpha_1 + \alpha_2 - \alpha_1\alpha_2}{K} + D(\alpha_1 + \alpha_2 - 1 - \frac{3}{4}\alpha_1\alpha_2) - \frac{\alpha_1}{2B_1} - \frac{\alpha_2}{2B_2} + \frac{3\alpha_1\alpha_2}{8}\left(\frac{1}{B_1} + \frac{1}{B_2}\right) > 0$$

Hasonlóan feírható a (11) egyenlőtlenség is, ennek közlésétől az általános alak bonyolultsága miatt eltekintünk.

A korábban meghatározott paraméterértékekkel a (8), (10) és (11) egyenlőtlenségek a következőképpen írhatók fel:

$$0 < \frac{14}{K} + D - 121.92$$

$$0 < \frac{0.8}{K} - 0.125D - 37.097$$

$$0 < (0.16K^2)D^2 + (16.773K^2 - 0.4K)D + 0.24 - 22.598K - 781.538K^2$$

Az első két egyenlőtlenségből D -re alsó illetve felső korlát adódik, amelyek egyidejűleg pontosan akkor állhatnak fenn, ha

$$K < \frac{20.4}{418.69} = 0.04872$$

Numerikusan megvizsgáltuk a rendszer viselkedését a $K \leq 1$ tartományban, $-100 \leq D \leq 0$ mellett, egyelőre még a nulla alsó korlát bevezetése nélkül.

A modellezett piaci ár-idősor $K = 1.0$ mellett D teljes tartományában széttartó oszcillálást mutat, igen nagy abszolút értékű, 10^{300} nagyságrendű pozitív és negatív értékekkel. K csökkentésével a modell viselkedésének jellege változatlan marad, csupán annyi a változás, hogy kisebb K értékeknél a pozitív és negatív értékek távolodása valamivel lassabb.

A viselkedés jellegében az első változás $K = 0.02$ mellett észlelhető, ahol a legnagyobb D értékek mellett a divergens viselkedés $D > -2$ mellett egyre kisebb oszcillációkat kezd felvenni. Ezután K további csökkentése hatására az oszcilláció távolságának csökkenése egyre kisebb D értékek mellett elkezdődik, és a 0-hoz közeli értékek esetén a rendszer konvergenssé válik.

Az 1. ábra $K = 0.015$ és $D = -90$ mellett mutatja az ár idősor alakulását. Jól látható, hogy az ár mintegy 100 időperiódus eltelte után a $+30000$ és a -30000 értékek közötti oszcillálásra áll be. Ugyanez a K érték $D = -80$ esetén már konvergens viselkedést eredményez, ahogyan az a 2. ábrán látható.

A 3. ábra a $K = 0.02$ értékhez tartozó *bifurkációs diagramot*² mutatja, amelyen jól látható, hogy $D < -56$ mellett a rendszer oszcillál egy pozitív és egy negatív érték közt, $D > -56$ mellett viszont a fixponthoz konvergál.

K további csökkentésével egyre kisebb D -től kezdődően konvergens a rendszer, és $K = 0.01$ esetén már a teljes $-100 \leq D < 0$ tartományban konvergenssé válik (4. ábra).

Ezek után megvizsgáltuk a rendszer viselkedését a nulla alsó korlát bevezetése mellett is. A korlát bevezetésének szükségességét az indokolja, hogy azon K és D értékpárok esetén, amikor a rendszer oszcillált, az alsó felvett érték mindenütt negatív volt, ami piaci árak esetében nehezen értelmezhető.

²A bifurkációs diagram az az ábra, amely megmutatja, hogy a paraméter különböző értékei mellett hosszabb távon, másképpen fogalmazva, egy kezdeti átmeneti szakasz után hogyan viselkedik a rendszer. Úgy készül, hogy azt a szakaszt, amelyben a paraméter lehetséges értékei változhatnak, egyenletesen felosztják nagyon sok részre. Minden osztópontban külön-külön elvégzik a következőket. A dinamikus rendszerrel megtesznek egy bizonyos számú lépést, elegendően sokat ahhoz, hogy az említett átmeneti szakaszt biztosan túlhaladják. Majd megtesznek újabb k lépést, és a rendszer itt elért értékeit rajzolják ki pontokkal a síkon úgy, hogy ha a paraméter értékét a vízszintes tengelyen ábrázolják, akkor a megőrzött k pontot az adott paraméterértékhez tartozó függőleges egyenesre teszik. k megválasztása attól függ, hogy mennyire finom részleteket lehet kinyomtatni.

Az alsó korlát bevezetése a korlátok nélküli modellváltozathoz képest jóval komplexebb viselkedést mutatott. Ismét $K = 1.0$ volt a vizsgálatok kiindulópontja. Ezen értéknél a teljes $-100 \leq D < 0$ tartományban kaotikus viselkedést mutatott a rendszer (5. ábra), és K csökkentése mellett is megmaradt ez a jelleg, csupán az ár értéke két egyre jobban elkülöníthető sávra válik szét (6. ábra). $K = 0.08$ alatt azonban a kaotikusság megszűnik, és egy kettes ciklus alakul ki, amelynek alsó értéke maga az előírt nulla alsó korlát (7. ábra). K további csökkentésével aztán először a 0-hoz közeli D értékeknél, majd később egyre kisebb D -nél is megjelenik ismét a kaotikus viselkedés (8. ábra), majd miután a teljes D vizsgálati tartomány kaotikussá vált, K további csökkentése a kaotikus ár idősor értékek határozott sávokra való szétválását okozza (9. ábra). Ezek a sávok aztán K további csökkentésével egyetlen ponttá olvadnak össze, azaz a kaotikus jelleg ciklikus viselkedéssé válik (10. ábra).

A 11. ábra a bifurkációs diagramok mellett egy példát mutat a tényleges, kaotikus idősor menetére is, $K = 0.05$, $D = -50$ mellett.

3.2 Az extrapolatív eset

Az extrapolatív modell alapegyenlete a következő lesz:

$$p_{l+1} = K(d + \sum_{k=1}^N \frac{b_k}{2B_k}) + (1 + KD)p_l - (\sum_{k=1}^N \frac{\alpha_k K}{2B_k})p_{l-1} - (\sum_{k=1}^N \frac{(1 - \alpha_k)K}{2B_k})p_{l-2} \quad (12)$$

A megfelelő karakterisztikus egyenlet:

$$\lambda^3 - (1 + KD)\lambda^2 + (\sum_{k=1}^N \frac{\alpha_k K}{2B_k})\lambda + \sum_{k=1}^N \frac{(1 - \alpha_k)K}{2B_k} = 0 \quad (13)$$

A stabilitás feltételei az adaptív esethez hasonlóan írhatók fel itt is, az ott közölt öt egyenlőtlenséggel. Ezek a következőképpen alakulnak:

$$0 < -KD + K \sum_{k=1}^N \frac{1}{2B_k} \quad (14)$$

$$D > \sum_{k=1}^N \frac{2\alpha_k - 1}{2B_k} - \frac{2}{K} \quad (15)$$

$$D < \frac{2}{K} - \sum_{k=1}^N \frac{3 - 2\alpha_k}{2B_k} \quad (16)$$

$$D > \frac{-4}{K} + \sum_{k=1}^N \frac{-3 + 2\alpha_k}{2B_k} \quad (17)$$

$$K^2 \left(\sum_{k=1}^N \frac{1 - \alpha_k}{B_k} \right)^2 + 2K^2 D \left(\sum_{k=1}^N \frac{1 - \alpha_k}{B_k} \right) + 2K \sum_{k=1}^N \frac{1}{B_k} - 4 > 0 \quad (18)$$

Látható, hogy a (14) feltétel minden $D < 0$ és $K > 0$ esetén teljesül. Egyszerű számolással az is látható, hogy a (15) egyenlőtlenség mindig erősebb alsó korlátot ad D -re, mint a (17). Eszerint rendszerünk pontosan akkor rendelkezik stabil fixponttal, ha (15), (16) és (18) egyenlőtlenség teljesül. (15) és (16) összevetéséből azonnal adódik K -ra egy szükséges feltétel, eszerint:

$$\frac{4}{K} > \sum_{k=1}^N \frac{1}{B_k}$$

azaz

$$K < \frac{4}{\sum_{k=1}^N \frac{1}{B_k}} \quad (19)$$

Az adaptív esethez hasonlóan itt is a két termelő esetét vizsgáljuk. Vezessük be az egyszerűbb kezelésmód érdekében a következő jelöléseket:

$$A = \frac{\alpha_1}{B_1} + \frac{\alpha_2}{B_2}$$

$$B = \frac{1}{B_1} + \frac{1}{B_2}$$

A (18) egyenlőtlenség kis átrendezésével:

$$D > \frac{2}{K^2(B-A)} - \frac{B}{K(B-A)} - \frac{1}{2}$$

Ezt összehasonlítva (16) által D -re adódó felső korlattal a következő szükséges feltétel adódik:

$$\frac{2}{K^2(B-A)} - \frac{B}{(B-A)K} - \frac{1}{2} < \frac{2}{K} - \frac{3B}{2} + A$$

ahonnan kis átalakítással az alábbi szükséges feltételt kapjuk:

$$0 < K^2(2A - 3B + 1) + K\left(4 + \frac{2B}{B-A}\right) - \frac{4}{B-A} \quad (20)$$

Konkrét számpéldánk adatait behelyettesítve:

$$0 < -474.399K^2 + 8.249K - 0.0348$$

A jobboldalon álló másodfokú függvény pontosan a gyökei közt lesz pozitív. Ezek: $K_1 = 0.0102$ és $K_2 = 0.0072$ lesznek. Ezáltal K -ra egy erős szükséges feltételt kaptunk, melyre ellenőrizhető, hogy teljesülhet-e valamely negatív D -re a többi feltétel, azaz lehet-e a rendszernek stabil fixpontja.

Szintén numerikus vizsgálatokkal elemeztük a modell instabil viselkedését, először a nulla alsó korlát bevezetése nélkül. Ismét $K = 1$ -től kezdtük a vizsgálatot, és az adaptív esettől eltérő módon itt minden $K \leq 1$ mellett oszcilláló viselkedést tapasztaltunk. Az oszcillálás azonban nem két érték, hanem több pozitív és több negatív érték közt játszódott le (12. és 13. ábra).

A nulla alsó korláttal kiegészített modell itt is rendkívül változatos viselkedést produkált. A $0.04 < K \leq 1$ tartományban mindenütt a nulla és egy pozitív érték közt oszcillált a rendszer (14. ábra). $K = 0.039$ esetén $D = -5$ értéknél bifurkáció történik, és $-5 < D < 0$ esetén három érték közt oszcillál az idősor. (15. ábra). K további csökkentésével a 0-hoz közeli D értékeknél kaotikus sáv jelenik meg, ami egyre kisebb D értékekre is áttérjed (16., 17., 18. ábra). Aztán a kaotikus sáv szélessége csökkenni kezd, majd $K = 0.0174$ és $K = 0.0173$ közt a még kaotikus viselkedés stabil fixponttá válik (19. és 20. ábra). Illusztrációképpen ismét megmutatjuk, milyen az idősor menete a még éppen kaotikus $K = 0.0174$ mellett $D = -63$ esetén (21. ábra).

4. Következtetések

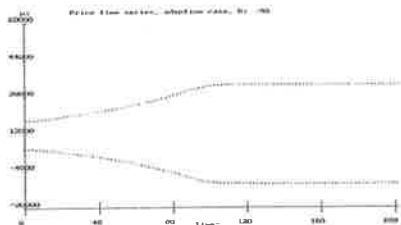
A fentiek szerint összességében megállapítható, hogy:

- A 0 alsó korlát bevezetése a viselkedést megváltoztatta, biztosította az árak nemnegativitását, de egyúttal kaotikus viselkedést is okozott.
- $K \geq 1$ mellett az árak gyakran irreálisan magasak, nemcsak a divergens idősorokban, hanem a kettes ciklusos oszcillációk mellett is (esetenként 1 millió Ft/t értéket meghaladó árak sem ritkák).
- $K < 1$ mellett K értékének igen kis megváltoztatása alapvetően megváltoztatja a modell viselkedésének jellegét (ahogy azt a 19. és a 20. ábra mutatja); hasonló érzékenységet tanúsít a modell a D értékének változtatására is.
- Az adaptív modellváltozat és az extrapolatív modellváltozat viselkedése lényegesen eltért, ami a termelői árbecslési mód helyes megválasztásának jelentőségét húzza alá. Ilyen lényeges eltérés volt például, hogy a nulla alsó korlátba az adaptív modell minden olyan esetben beleütközött, amikor a rendszer nem fixponthoz konvergált, míg az extrapolatív modellben kaotikus viselkedés is előfordult csupa pozitív árral (19. ábra).

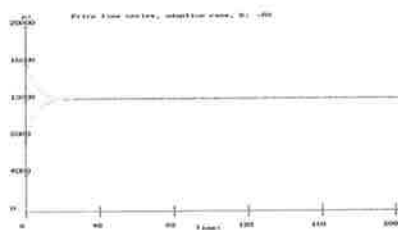
Mindezek alapján megállapítható, hogy az alkalmazott, a gyakorlat szempontjából realiztikus feltételezésekkel élve, a statisztikailag becsült, és így a valóságtól nem túl eltérő paraméter-értékek mellett a piac viselkedése mind az adaptív, mind az extrapolatív viselkedést feltételezve kaotikus is lehet. Mindez alapvetően eltér a nulla alsó korlát bevezetése nélküli változattól, ahol fixpont és oszcilláló divergencia volt a két jellemző viselkedési forma, és a lineáris dinamikus rendszerek elméletéből ismeretes, hogy nem is fordulhat elő kaotikus viselkedés.

Szükségesnek tartjuk kiemelni, hogy az alsó korláttal ellátott modell kaotikus, illetve fluktuáló jellege nem valamely külső sokkhatás, vagy szokatlan környezeti körülmény miatt alakul ki, hanem a piac belső jellegzetessége, alapvető tulajdonsága. Emiatt a burgonyapiac stabilizálására törekvő intézkedéseknek is igen körültekintően kell vizsgálniuk ezen piac belső szerkezetét és dinamikáját. Feltűnően érzékenyen reagált a modell viselkedése a K paraméter értékére, ami jelzi ezen paraméter helyes és minél pontosabb meghatározásának fontosságát. További lehetőségként vetődik fel, hogy a K paraméter esetleges külső befolyásolásával a kaotikus, vagy erősen oszcilláló viselkedés szabályozható, a kilengései mérsékelhetőek, ha teljesen meg nem is szüntethetőek. Mindez a termelők szempontjából igen jelentős lehet, hiszen a bevételekben várható szélsőséges ingadozások helyett kedvezőbb, kiszámíthatóbb bevételt érhetnek el, ami az időjárás bizonytalanságának amúgy is kitett termelők gazdálkodási feltételeit javíthatja.

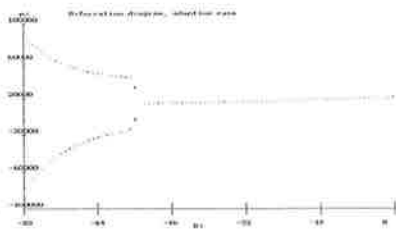
A fent bemutatott elemzés matematikai érdekessége, hogy egy alapvetően lineáris modellszerkezet minimális megváltoztatása, jelen esetben a nulla alsó korlát bevezetésével szakaszonként lineárisra tétele alapvetően megváltoztathatja a modell viselkedését, azaz elégséges feltétel lehet kaotikus viselkedés kialakulására is.



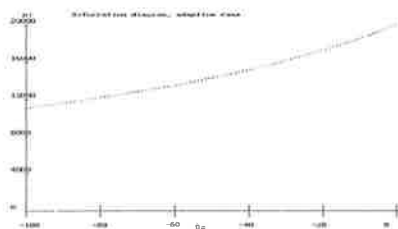
1. ábra: Adaptív modell korlát nélkül,
 $K=0.015$, $D=-90$, az idő függvényében.



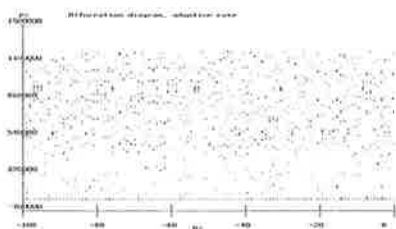
2. ábra: Adaptív modell korlát nélkül, $K=0.015$, $D=-80$, ár az idő függvényében



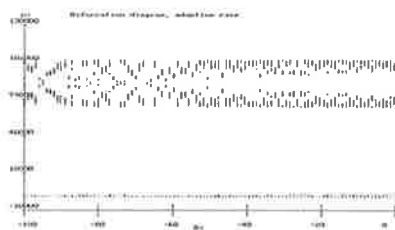
3. ábra: Adaptív modell, korlát nélkül, $K=0.02$, bifurkációs diagram



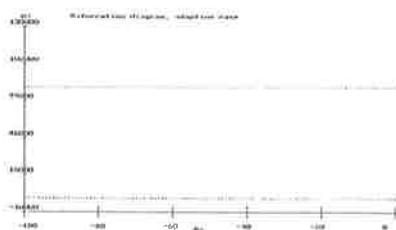
4. ábra: Adaptív modell, korlát nélkül, $K=0.01$, bifurkációs diagram



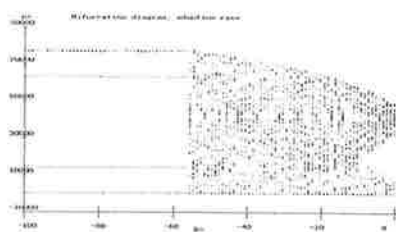
5. ábra: Adaptív modell, alsó korláttal, $K=1.0$, bifurkációs diagram



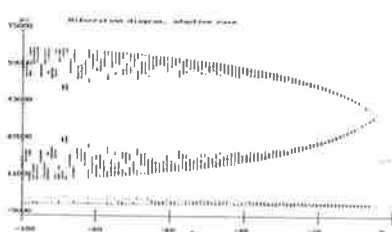
6. ábra: Adaptív modell, alsó korláttal, $K=0.08$, bifurkációs diagram



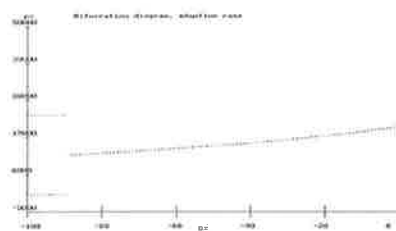
7. ábra: Adaptív modell, alsó korláttal, $K=0.075$, bifurkációs diagram



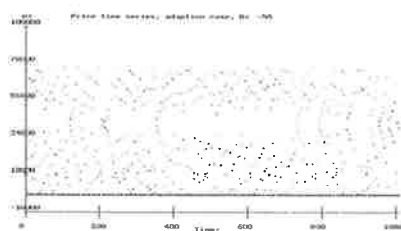
8. ábra: Adaptív modell, alsó korláttal,
 $K=0.05$, bifurkációs diagram



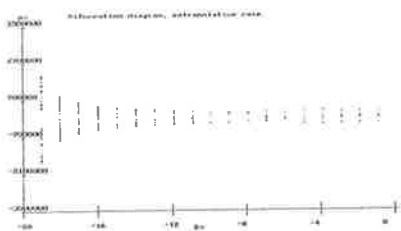
9. ábra: Adaptív modell, alsó korláttal,
 $K=0.041$, bifurkációs diagram



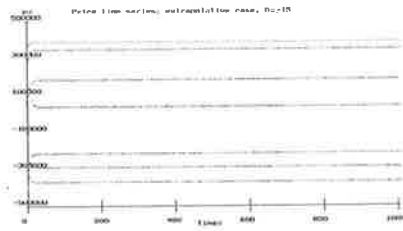
10. ábra: Adaptív modell, alsó korláttal,
 $K=0.015$, bifurkációs diagram



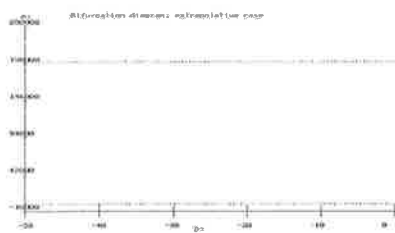
11. ábra: Adaptív modell, alsó korláttal,
 $K=0.05$, $D=-50$, ár az idő függvényében



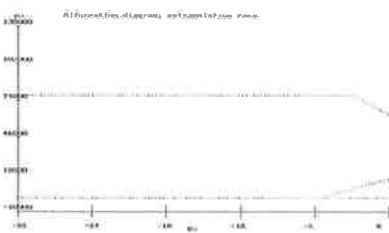
12. ábra: Extrapolatív modell korlát nélkül,
 $K=0.1$, bifurkációs diagram



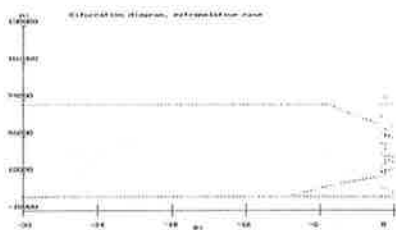
13. ábra: Extrapolatív modell korlát nélkül,
 $K=0.1$, $D=-15$, ár az idő függvényében



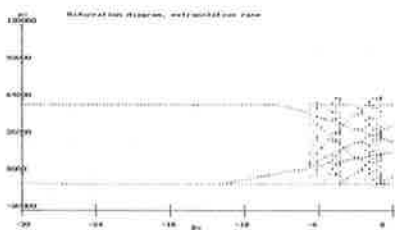
14. ábra: Extrapolatív modell alsó korlátal,
 $K=0.1$, bifurkációs diagram



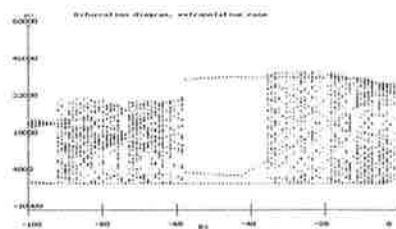
15. ábra: Extrapolatív modell alsó korlátal,
 $K=0.039$, bifurkációs diagram



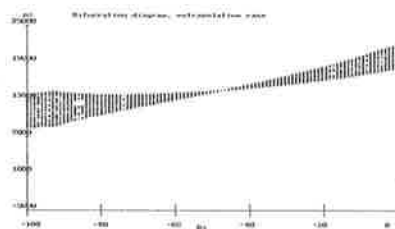
16. ábra: Extrapolatív modell alsó korlátal,
 $K=0.035$, bifurkációs diagram



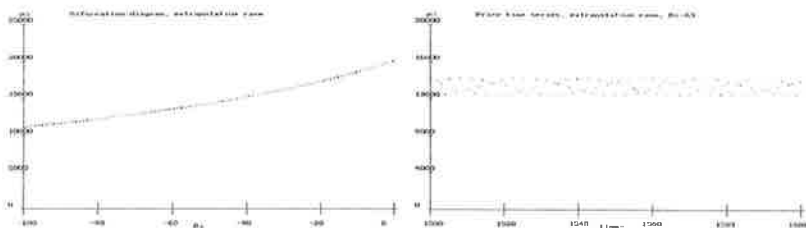
17. ábra: Extrapolatív modell alsó korlátal,
 $K=0.03$, bifurkációs diagram



18. ábra: Extrapolatív modell alsó korlátal,
 $K=0.02$, bifurkációs diagram



19. ábra: Extrapolatív modell alsó korlátal,
 $K=0.0174$, bifurkációs diagram



20. ábra: Extrapolatív modell alsó korláttal,
 $K=0.0173$, bifurkációs diagram

21. ábra: Extrapolatív modell alsó korláttal,
 $K=0.0174$, $D=-63$, ár az idő függvényében

Irodalom

1. Benhabib, J (ed.): *Cycles and chaos in economic equilibrium*, Princeton University Press, Princeton, 1992.
2. Bessenyei I., "Árbeccési módszerek értékelése Szidarovszky és Molnár diszkrét dinamikus termelői-fogyasztói modellje alapján", *Sigma*, 27(1996) 35–43.
3. Molnár S., Szidarovszky F., "Egy diszkrét dinamikus termelői-fogyasztói modell stabilitásáról", *Sigma*, 25(1994) 207–219.
4. Vizvári, B., Bacsi, Zs., Kolumbán G., Chaotic Dynamics of Agricultural Markets: the Case Study of the Hungarian Potato Market, *RUTCOR, Rutgers University, Research Report*, RRR 31–96.

INSTABILITY AND CHAOTIC BEHAVIOUR IN A DISCRETE DYNAMIC PRODUCER-CONSUMER MODEL BY MOLNÁR AND SZIDAROVSKY

The present paper analyses the behaviour of a discrete dynamic producer-consumer model defined by Molnár and Szidarovszky (1994), and focuses on the unstable region of the adaptive and the extrapolative model versions, assuming that the market price changes proportionally to the level of disequilibrium, that is, the difference between the demanded and the supplied product amount. A lower limit was attached to the market price in the model, namely, the price was limited to nonnegative values. Due to this limit the model structure becomes nonlinear instead of the original linear structure. This simple structure generates very complicated behaviour, including chaotic behaviour, and shows great sensitivity to the parameter value K , which is the coefficient associated with the supply-demand difference in the price formation.

EGY IMPLICIT LESZÁMLÁLÁSON ALAPULÓ ÚJ ERŐFORRÁS KIEGYENLÍTŐ ELJÁRÁS¹

CSÉBFAI GYÖRGY – PETROS KONSTANTINIDIS

Janus Pannonius Tudományegyetem – Technological Educational Institution
of Athens

Dolgozatunkban egy új implicit leszámolásán alapuló egzakt erőforrás kiegyenlítő eljárást ismertetünk. Az eljárás tevékenységek tartalékidőn belüli késleltetésével olyan ütemtervet keres, amelyben az erőforrás hisztogramok alakja megfelel a kívánatos konkáv alaknak. Az eljárás kiindulási adatait a kritikus út módszere szolgáltatja. Tapasztalataink az mutatják, hogy kis és közepes méretű problémák esetében az egzakt megoldások előállítása nem igényel komolyabb számítástechnikai háttérrel.

1. Bevezetés

Projektek erőforrás felhasználásával kapcsolatos ütemezési problémák két alapvető csoportba oszthatók, ennek megfelelően erőforrás hozzárendelési, illetve erőforrás kiegyenlítési problémáról beszélhetünk. Mindkét probléma kiindulási alapját a kritikus út módszerével kapott legkorábbi tevékenység ütemezésnek megfelelő ütemterv alkotja.

Hozzárendelési problémáról van szó, ha a rendelkezésre álló erőforrások korlátozottsága miatt a kritikus út módszerével kapott legkorábbi ütemterv nem valósítható meg. Ekkor az erőforrás felhasználási konfliktusok feloldásával, vagyis a kritikus és a nem kritikus tevékenységek ütemezésének késleltetésével olyan ütemtervet kell keresünk, amely kielégíti az erőforrás korlátokat és amely a kritikus út módszerével kapott befejezési időpontot minimális mértékben növeli. A jól ismert nehéz probléma szakirodalma igen széles körű, számtalan heurisztikus és jó néhány egzakt módszert dolgoztak ki a probléma megoldására.

Az erőforrás kiegyenlítési probléma esetében a kritikus út módszerével kapott ütemtervek megvalósítását erőforrás korlátok nem akadályozzák, de megoldandó problémát jelentenek az erőforrás felhasználásban mutatkozó ingadozások. Ekkor kiindulva a legkorábbi ütemtervből, a nem kritikus tevékenységek ütemezésének késleltetésével egy olyan ütemtervet keresünk,

¹Érkezett: 1998. február 4.

amelyben az erőforrás felhasználási hisztogramok alakja a lehető legjobban megközelíti a kívánatos alakot. Egy erőforrás esetében a feladat célfüggvényét hagyományosan az erőforrás felhasználás szórásnégyzeteként (az átlag felhasználástól való négyzetes eltérésként) definiáljuk.

A probléma megoldására számos heurisztikus eljárást dolgoztak ki. Az eljárások közös vonása, hogy kiindulva a legkorábbi ütemtervből, valamilyen ökölszabály alkalmazásával bizonyos tevékenységek ütemezését készletetik. Két jellegzetes példaként Harris (1990) és Konstantinidis (1998) módszerét említhetnénk.

A probléma egzakt megoldására szolgáló módszer szinte alig van. Jellegzetes irányzatokat képvisel Ahuja (1976) explicit leszámítást alkalmazó módszere, Valadares Taveres (1987) eljárása amely a kívánatos hisztogram alakot folytonos időfüggvényként kezeli, Easa (1989) egész értékű optimalizáláson alapuló modellje, illetve a Bandeloni és társai (1994) által kifejlesztett egy erőforrást kezelő, dinamikus programozáson alapuló módszer.

A cikk egy új, az implicit leszámítás módszerén alapuló egzakt algoritmust ismertet az erőforrás kiegyenlítési probléma megoldására. Az eljárás az előzőekben említett eljárásokhoz képest továbblépést jelent azáltal, hogy az explicit leszámításnál hatékonyabb, nem követeli meg a kívánatos alak folytonos időfüggvényként történő megadását, nem igényli számításigényes egész értékű programozási feladat megoldását és semmilyen formában nem korlátozza az ütemtervben szereplő erőforrások számát.

Az algoritmus új eleme, hogy a kívánatos hisztogram alaknak a lapos konkáv alakot tekinti, tehát szakít a négyzetes eltérés típusú célfüggvények hagyományával. A váltás mögött az a felismerés áll, hogy egyrészt

- a gazdaságilag nehezen megfogható ingadozás minimalizálás mögött lényegében mindig a jól számszerűsíthető állásidő minimalizálási törekvés húzódik meg, annak ellenére, hogy a minimális ingadozású és a minimális állásidőjű ütemezés nem azonos, másrészt
- gazdasági szempontból a mesterkéltséggel átlagos erőforrás felhasználásnál a maximális felhasználás lényegesen többet mond, a menedzsment a maximális felhasználás minimalizálásában, vagyis az erőforrás hisztogramok összelapításában érdekelt.

A módszertani felvetés lényegében a hozzárendelési probléma és kiegyenlítési probléma összekapcsolását jelenti. Rögzítve a projekt befejezési időpontját, erre elméletileg három lehetőségünk van:

1. Megoldjuk a kiegyenlítési problémát, majd a megoldások halmazából kiválasztjuk azokat, amelyek kielégítik az erőforrás korlátokat.

2. Megoldjuk az erőforrás hozzárendelési problémát, majd a megoldások halmazán megkeresünk konkáv hisztogramokkal jellemezhető megoldásokat.
3. A két problémát egyesítjük, vagyis olyan ütemtervet keresünk, amelyben minden erőforrás hisztogram konkáv és a megadott korlátan belül marad.

Azt, hogy a három megközelítés közül melyik a legjobb, szinte lehetetlen megválaszolni, mivel konkrét feladatok esetében, a megoldáshalmazok függvényében, igen eltérő feleleteket kaphatunk (Csébfalvi és Konstantinidis (1998)). Az értékelést tovább nehezíti, ha a kiegyenlítési problémát *liberalizáljuk*, vagyis közelítőleg konkáv alakokat is megengedünk.

Jelen keretek között csak az első esettel, vagyis a kiegyenlítési probléma megoldásával foglalkozunk. A második eset az erőforrás hozzárendelési probléma összes lehetséges megoldásának ismeretét igényli, így át kell fogalmaznunk az erőforrás hozzárendelési problémát, hiszen ezen hagyományosan csak az első megoldás megkeresését értjük (Csébfalvi (1998)). A harmadik eset, azaz a két szempont szerinti egyidejű keresés, módszertanilag teljes egészében új kutatási területet jelent, azzal a kecsegtető lehetőséggel, hogy a keresési szempontok összekapcsolása a keresési fa méreteit csökkenti.

2. Az erőforrás kiegyenlítés módszere

Álljon a projekt $N + 2$ tevékenységből, és sorszámozzuk a tevékenységeket 0-tól $N + 1$ -ig. A szokásos módon jelölje a 0-adik és az $N + 1$ -edik ún. mesterséges tevékenység a projekt egyedi kezdetét és végét. Jelölje D_i az i -edik ($i = 1, 2, \dots, N$) tevékenység időtartamát. A tevékenységek közötti függőségi viszonyokat vég-kezdet típusú logikai kapcsolatokkal jellemezzük. A projekt ábrázolásakor a "tevékenységek csúcspontok, a kapcsolatokat élek" konvenciót alkalmazzuk. Feltételezzük, hogy az ütemezési probléma minden adata egész értékű.

Jelölje E_i , illetve L_i az i -edik ($i = 1, \dots, N$) tevékenység legkorábbi, illetve legkésőbbi kezdési időpontját. Legyen $F_i = L_i - E_i$ az i -edik ($i = 1, \dots, N$) tevékenység tartalékideje. Jelölje R az erőforrások számát, Q_r az r -edik ($r = 1, 2, \dots, R$) erőforrásból időegységenként rendelkezésre álló mennyiséget, illetve $U_{i,r}$ ($i = 1, 2, \dots, N$; $r = 1, 2, \dots, R$) az i -edik tevékenység időegységre eső erőforrás igényét az r -edik erőforrásból. Legyen T a projekt rögzített befejezési időpontja. Jelölje

$$S = \{\{t_1, s_1\}, \{t_2, s_2\}, \dots, \{t_n, s_n\}\}, t_i \in \{1, 2, \dots, N\}, n \leq N \quad (1)$$

egy adott ütemtervhez tartozó késleltetések halmazát, ahol t_i a tevékenység sorszáma, s_i a tevékenység késleltetése időegységekben mérve, amelyre igaz, hogy

$$s_i > 0, s_i \leq F_{t_i}. \quad (2)$$

A késleltetés a tevékenység *legkorábbi kezdési időpontjához képest* jelent késleltetést, így a legkorábbi ütemtervre $S = \{\}$.

Nevezzük az S halmazban szereplő tevékenységeket *kötött* tevékenységeknek. Egy adott ütemtervből csak a *szabad* (nem kötött) tevékenységek késleltetésével konstruálhatunk újabb ütemterveket.

Jelölje $C_{tr} = C_{tr}(S)$ egy adott S késleltetés halmazzal jellemezhető ütemterv esetében a t -edik ($t = 1, 2, \dots, T$) időpontban a tevékenységek együttes erőforrás felhasználását az r -edik ($r = 1, 2, \dots, R$) erőforrásból.

1. Definíció. *Nevezzük az r -edik erőforrás hisztogram $[a, b]$ időintervallumban eső részhalmazát völgynek, ha $b - a > 1$,*

$$\text{Min}(C_{ar}, C_{br}) > C_{kr} \text{ minden } k \in [a + 1, b - 1] \quad (3)$$

értékre és ahol az $[a, b]$ időintervallum olyan nagy, amilyen csak lehetséges.

Jellemezzük a völgyeket az $\{a, b, r, p\}$ adatokkal, ahol

$$p = \sum_{k=a+1}^{b-1} \text{Max}(\text{Min}(C_{ar}, C_{br}) - C_{kr}, 0) \quad (4)$$

a völgy nagysága, vagyis az állásidők összege.

Jellemezzük magát az ütemtervet az időtengelyen balról jobbra, illetve azon belül az erőforrásokon növekvő sorrendben haladva a völgyek

$$V = \{\{a_1, b_1, r_1, p_1\}, \{a_2, b_2, r_2, p_2\}, \dots, \{a_v, b_v, r_v, p_v\}\}, v = |V| \quad (5)$$

halmazával, illetve a

$$P = \begin{cases} \sum_{i=1}^v p_i, & \text{ha } v > 0; \\ 0, & \text{ha } v = 0 \end{cases} \quad (6)$$

mennyiséggel, mint *értékelési kritériummal*.

Ha egy erőforrás hisztogram nem tartalmaz völgyeket, vagyis alakja konkáv, akkor az adott erőforrás igénybevételét leállások nem szakítják meg. Ha a kívánatos alak a tevékenységek késleltetésével elérhető, akkor a projekt által nem hasznosított erőforrás kapacitás a projekt *elején és végén* jelentkezik, megteremtve a másirányú hasznosítás lehetőségét. *Több projekt*es alkalmazási környezetben például, az így adódó kapacitást más projektekhez rendelhetjük.

Hangsúlyoznunk kell, hogy az $\{r_i, p_i\}$, $i = 1, 2, \dots, v$ értékekből kiindulva sokféle értékelési kritérium konstruálható, azonban ezek taglalásától a jelen keretek között eltekintünk. A $P = P(S)$ választás, mint az állásidők összege, a legegyszerűbb esetet jelenti, mivel a különböző erőforrásokat összemossa. Az erőforrások költségfüggvényének bevezetésével a kívánatos cél sokkal *árnyaltabban* is megfogalmazható.

Ezen túlmenően meg kell jegyeznünk, hogy különösen több erőforrásos projektek esetén a *minden erőforrás dimenzióban kiegyenlített* ütemezés — a technológiai kötöttségek miatt — nem mindig érhető el. A közölt algoritmus, az érdemi elemek megtartásával, illetve egy megengedett $P_{\text{tűrés}}$ bevezetésével továbbfejleszhető. Az általánosított algoritmus kidolgozása folyamatban van és a későbbiekben közöljük.

Az ismertetendő keresési eljárásban a *legelső völgy* kiemelt szerepet játszik, mivel az algoritmus minden lépésben a kiválasztott *legígéretesebb* ütemterv *legelső* völgyének feltöltésére törekszik. A legígéretesebb ütemterv alatt azt az ütemtervet értjük, amelynek az *állásideje a legkisebb* és ezen belül azt, amelyben *legelső völgy a lehető legkésőbb* található. A másodlagos értékelési kritérium mögött az a feltételezés áll, hogy a minimális állásidővel rendelkező ütemtervek közül valószínűleg az áll *"legközelebb"* a probléma megoldásához, amelyre vonatkozóan a másodlagos értékelési kritérium fennáll.

2. Definíció. Akkor mondjuk, hogy egy $S_{r_{ij}}$ késleltetés halmazzal jellemzett ütemterv *legelső*, az r -edik hisztogram $[a, b]$ intervallumába eső völgye feltölthető, ha van olyan

$$S_{uj} = S_{r_{ij}} \cup S_{\text{feltöltött}} \quad (7)$$

késleltetés halmaz, hogy az ehhez tartozó ütemterv minden erőforrás hisztogramja konkáv az $[1, b]$ intervallumban.

Az *implicit leszámítás* logikájának megfelelően az eljárás egy keresési fát épít fel, amelynek gyökerét a kritikus út módszerével kapott legkorábbi ütemterv alkotja, amelyről feltesszük, hogy nem konkáv az $[1, T]$ időintervallumban. Az algoritmust az *összes* kiegyenlített megoldás keresésére alkalmas formában fogalmazzuk meg.

Köztudott, hogy az implicit leszámítás módszerének számítástechnikai hatékonyságát (a keresési fa méreteit) alapvetően az befolyásolja, hogy viszonylag egyszerű és gyors próbák segítségével milyen mértékben sikerül a fa bizonyos részeit a további vizsgálatokból kizárni. Hatékony *metsszési szabályok* nélkül az implicit leszámítás hatékonysága az explicit leszámítás szintjén mozog.

A javasolt eljárás a *kiegyenlítési problémát feltöltési problémák sorozatára* vezeti vissza. A lényegét tekintve, a feltöltési probléma és az eredeti probléma

azonos. A feltöltési probléma azonban, extrém esetektől eltekintve, jóval *kisebb méretű*, megoldása gyors, ezért alkalmas arra, hogy segítségével a keresési fa felesleges ágait levágjuk.

Ha egy ütemtervben egy völgyről bizonyítható, hogy nem tölthető fel, akkor az ütemtervet a további vizsgálatokból kizárhatjuk. Meg kell jegyeznünk, hogy a bizonyítás nem feltétlenül jelenti az összes feltöltési lehetőség explicit leszámllását, mivel az esetek egy részében az explicit leszámllás egy egyszerű és gyors próbával helyettesíthető.

Jelölje egy adott S késleltetés halmaz esetében K_{tr} illetve U_{tr} , ahol $t = 1, 2, \dots, T$ és $r = 1, 2, \dots, R$ az adott t időpontban az r -edik erőforrásra vonatkozóan a *minimális* illetve a *maximális* igényt, ahol a minimális (maximális) igényt az adott időpontban lehetséges tevékenység kombinációk összesített igényének *minimuma* (*maximuma*) adja.

Próba. Ha egy $\{a, b, r\}$ adatokkal jellemzett völgy esetében van olyan

$$h \in \{a + 1, a + 2, \dots, b - 1\} \quad (8)$$

időpont, hogy erre az időpontra vonatkozóan

$$U_{hr} < \text{Min}(K_{ar}, K_{br}), \quad (9)$$

akkor a völgy nem tölthető fel.

A keresési algoritmus lényegét az alábbiakban foglaljuk össze. A keresési fa csúcspontjait a $k = 0$ értékből kiindulva sorszámozzuk és az

$$\{S^{(k)}, P^{(k)}, V^{(k)}, I^{(k)}\} \quad (10)$$

értékekkel jellemezzük, ahol $S^{(k)}$ a k -edik csúcspont-hoz tartozó késleltetés halmaz, $P^{(k)}$ az állásidő nagysága, $V^{(k)}$ a völgyek halmaza, $I^{(k)}$ pedig a csúcspont indikátor változója,

$$I^{(k)} = \begin{cases} 0, & \text{ha a } k\text{-edik csúcspont még nincs kifejtve;} \\ 1, & \text{ha a } k\text{-edik csúcspont kifejtése már megtörtént;} \\ 2, & \text{ha a csúcspont optimális.} \end{cases}$$

Jelölje k_{\max} a legnagyobb csúcspont sorszámot (induláskor $k_{\max} = 0$). A keresési algoritmus lépései:

1. Minden egyes lépés a legígéretesebb, k_{\min} sorszámú csúcspont kiválasztásával kezdődik, amelynek keresési feltétele a következő:

$$P^{(k_{\min})} \rightarrow \text{Min}, \quad I^{(k_{\min})} = 0, \quad k_{\min} \in \{0, 1, \dots, k_{\max}\}. \quad (11)$$

Ha a keresés sikertelen, vagyis nincs olyan csúcspont, amelyet még nem vizsgáltunk meg, az eljárás befejeződik és az

$$O = \{ \{ S^{(k)}, P^{(k)}, V^{(k)}, I^{(k)} \} \mid I^{(k)} = 2, k \in \{0, 1, \dots, k_{\max}\} \} \quad (12)$$

halmaz adja a feladat optimális megoldásait. Természetesen az $O = \{ \}$ előállhat.

Ha a keresés eredményeképpen több csúcspontot kapunk, akkor ezek közül vesszük azt, amelyben legelső völgy a lehető legkésőbb található, mivel feltételezzük, hogy ez a csúcspont van a legközelebb a kiegyenlítési probléma megoldáshoz.

2. A kiválasztott csúcspont indikátor változójának értékét egyre változtatjuk, vagyis $I^{(k_{\max})} = 1$ és a csúcspontot kifejtjük.

A kifejtés során a kiválasztott csúcspont *legelső völgyének* feltöltési lehetőségei adják az

$$\{ S^{(k)}, P^{(k)}, V^{(k)}, I^{(k)} \}, \quad k = k_{\max} + 1, k_{\max} + 2, \dots, k_{\max} + k_{uj} \quad (13)$$

származtatott csúcspontokat, ahol

$$I^{(k)} = \begin{cases} 0, & \text{ha } P^{(k)} > 0; \\ 2, & \text{ha } P^{(k)} = 0. \end{cases} \quad (14)$$

A feltöltési lehetőségek közül *azonnal töröljük* azokat, amelyek esetében bizonyítható (lásd *Próba*), hogy valamelyik erőforrás dimenzióban olyan hisztogramot eredményeznének, amelynek van olyan völgye, amely nem tölthető fel.

Az új csúcspontok számának megfelelően módosítjuk k_{\max} értékét, vagyis $k_{\max} = k_{\max} + k_{uj}$ és visszatérünk az 1. lépésre.

3. Alkalmazási példa

A javasolt új erőforrás kiegyenlítő eljárás működését egy egyszerű alkalmazási példa bemutatásával szemléltetjük. A példa kiindulási adatait, illetve a kritikus út módszerének eredményeit az 1. táblázat tartalmazza. A feladat kapcsolatokkal kiegészített Gantt diagramját, illetve erőforrás hisztogramját a legkorábbi ütemezésnek megfelelően az 1. ábra szemlélteti. A feladatban csak egy erőforrás szerepel, amellyel kapcsolatban feltesszük, hogy igénybevétele nincs korlátozva. A projekt rögzített befejezési időpontja $T = 17$ időegység.

A *nem-kritikus*, vagyis késleltethető tevékenységek száma 5. A legkorábbi ütemezésnek megfelelő erőforrás hisztogram három völgyet tartalmaz. A kiindulási ütemterv esetében az értékelési kritérium értéke $P = 3 + 2 + 2 = 7$.

A lehetséges ütemezések száma 1860. *Explicit leszámolás* esetén ennyi esetet kellene megvizsgálnunk, hogy a konkáv hisztogrammal rendelkező eseteket kiválaszthassuk.

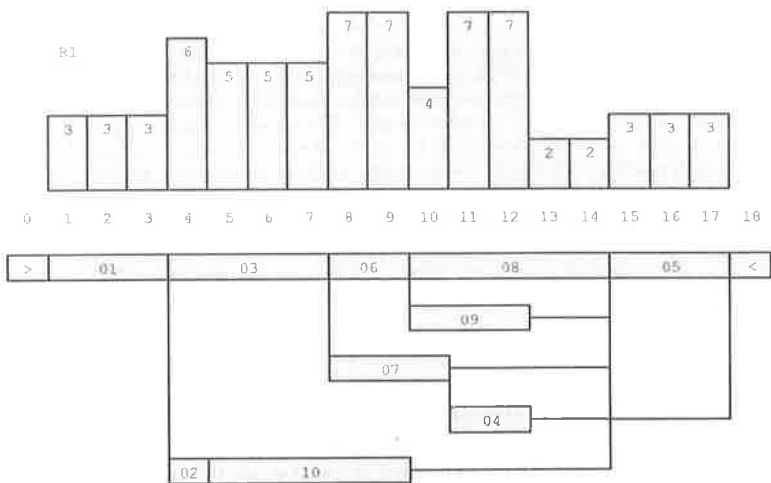
Az első völgy *"feltöltésében"* három változó játszik szerepet: $\{A02, A07, A10\}$. Az előzőekben ismertetett *vágási szabály alkalmazása nélkül* az első völgy feltöltéséhez 105 esetet kell megvizsgálnunk, amelyek közül 59 esetben kapunk a feltöltésre alkalmas késleltetés kombinációt. A *vágási szabály alkalmazásakor* a lehetséges késleltetési kombinációk száma 17, ezek közül 10 alkalmas feltöltésre. Mindkét esetben a megoldások között pontosan egy olyan késleltetési kombináció található, amely optimális megoldása a kiindulási problémának (2. ábra).

Az alkalmazott keresési stratégiának — *depth-first solution strategy* — megfelelően viszonylag kevés esetet kell megvizsgálnunk ahhoz, hogy az optimális megoldást megkapjuk (31 illetve 10). A további lépések (170 illetve 151) ahhoz szükségesek, hogy bizonyítsák, hogy az adott feladatnak nincsen más optimális megoldása. Az *összes* kiértékelt eset száma $31 + 170 = 201$ vágás nélkül, illetve $10 + 151 = 161$ vágással, ami jól szemlélteti az implicit leszámolás, illetve azon belül a javasolt vágási szabály előnyeit az explicit leszámoláshoz képest.

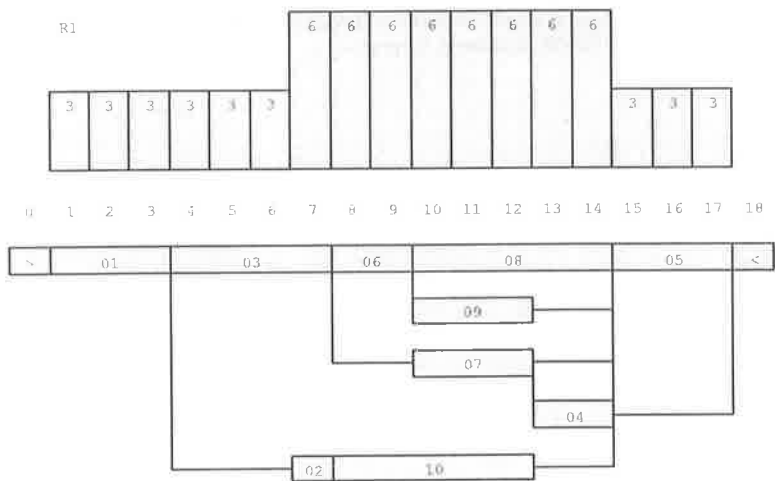
Megjegyezzük, hogy erőforrás korlátok beépítésével a kiértékelésre kerülő esetek száma tovább csökkenthető.

tev.	megelőző tevékenység	időtartam	erőforrás igény	legkorábbi ütemezés	legkésőbbi ütemezés
D00		1		0	0
A01	D00	3	3	1	1
A02	A01	1	3	4	9
A03	A01	4	3	4	4
A04	A07	2	4	11	16
A05	A07,A08,A09,A10	3	3	15	15
A06	A03	2	4	8	8
A07	A03	3	1	8	12
A08	A06	5	2	10	10
A09	A06	3	1	10	12
A10	A02	5	2	5	10
D11	A04,A05	1		18	18

1. táblázat. Kiindulási adatok és a kirikus út módszerének eredményei



1. ábra: A kritikus út módszerével adódó legkorábbi ütemezés



2. ábra: A kiegyenlítés eredményeképpen adódó ütemezés

OPTIMAL RESOURCE LEVELING USING IMPLICIT ENUMERATION

This paper presents a new exact approach for resource leveling based on implicit enumeration. The objective of the proposed approach is to schedule all the activities in the project to specific time periods within their float, such that the resource histograms meet the desired concave shape. The model, which is applied to an activity-on-node network, requires as input the scheduling results computed by the critical path method. Computational results indicate that the procedure provides exact solutions for small to medium size problems, requiring only modest computing facilities.

Irodalom

1. Valadares Tavares, L. (1987), Optimal Resource profiles for program scheduling, *EJOR* 29, 83–90
2. Bandelloni, M. et al. (1994), Optimal resource leveling using non-serial dynamic programming, *EJOR* 78, 162–177
3. Easa, S. M. (1989), Resource leveling in construction by optimization, *JCEM* 115 2, 302–316
4. Konstantinidis, P. (1998), A model to optimize project resource allocation by construction of a balanced histogram, *EJOR* 104, 559–571
5. Csébfalvi, G. (1998), A fast exact solution procedure for multiple resource constrained project scheduling, *PROC. APMOD '98, Limassol, Cyprus*
6. Csébfalvi, G. and Konstantinidis, P. (1998), A new exact resource balancing procedure for multiple resource constrained project scheduling problem, *PROC. APMOD '98, Limassol, Cyprus*.

EGY ÚJ KÜLSŐPONTOS ELJÁRÁS RÁCSOS TARTÓK TOPOLOGIAI TERVEZÉSÉRE¹

CSÉBFAI ANIKÓ – CSÉBFAI GYÖRGY

Janus Pannonius Tudományegyetem

A cikkben egy külsőpontos eljárást ismertetünk térbeli rácsos tartók optimális topológiájának meghatározására. A csomópontokból és rudakból álló térbeli rácsos tartó, mint szerkezeti modell számos tervezési probléma, épületszerkezet, hídszerkezet és repülőgép tervezési feladatok esetében fontos szerepet játszik. A topológiai optimálás során azt a minimális súlyú szerkezeti elrendezést keressük, amely adott terhelési viszonyok mellett kielégíti a szerkezet viselkedésével kapcsolatos korlátozó feltételeket. Az optimális topológia meghatározása — a szerkezeti modellből fakadóan — tipikusan nagyméretű, nemlineáris és nemkonvex optimálási feladat. A módszer alkalmazási lehetőségeit a szakirodalom egyik közismert teszt feladatával szemlélítjük.

1. Bevezetés

A topológiai optimálás, amelyen belül a csomópontok optimális topológiájának és a rúdelemek optimális keresztmetszeti értékeinek meghatározása egyidejűleg történik, az alapszerkezet — ground structure — feltételezésén alapuló szerkezet-optimálási feladatok csoportjába tartozik. A kiindulási alapszerkezetet úgy állítjuk elő, hogy a tervezési feladat függvényében előre meghatározott térbeli rácspontokat, mint a rácsos tartó csomópontjait, az összes lehetséges módon, rúdelemekkel kötjük össze. Az eljárás során elimináljuk az alapszerkezet felesleges — gazdaságtalan — rúdelemeit. A módszer elméleti alapjait és alkalmazási lehetőségeit Rozvany, Bendsøe és Kirsch (1995) monográfiája tárgyalja. Az első analitikus módszerek (Hemp (1973), Rozvany (1989)), amelyeket a "rácsszerű" kontinuum feladatok tulajdonságainak vizsgálatára dolgoztak ki, Michell (1904) munkáján alapulnak. Diszkrét modellek — pl. rácsos tartók — topológiai optimálási feladatainak numerikus módszereit ismertetik Kirsch (1989) és Ben-Tal (1993). Az elmúlt években dolgozták ki az ún. "homogenizáló" módszert (Bendsøe és Kikuchi (1988), Suzuki és Kikuchi (1991)), hangsúlyozva a topológiai optimálási feladatok

¹A cikk a T 018324 számú OTKA pályázat által támogatott kutatás keretén belül készült. Beérkezett: 1998. január 20.

gyakorlati szerepének fontosságát. Az a tény, hogy a topológiai optimalás területén még nincs akkora tapasztalatunk, mint a rögzített elrendezésű szerkezet-optimalási feladatok esetén, számos alapvető nehézséget vet fel, beleértve a megoldó módszerek sajátosságaiból fakadó problémákat is.

A térbeli rácsos tartók topológiai optimalása során az alábbi alapvető nehézségekkel kell szembenéznünk:

- a feladat tipikusan nagyméretű, mivel az alapszerkezet rúdelemeinek száma a rácsponatok számának kvadratikusan függvénye;
- a szerkezeti modell az optimalási eljárás során változik, mivel a zérus keresztmetszetű elemeket automatikusan töröljük az alapszerkezetből;
- az optimalási feladat nemlineáris és rendszerint nemkonvex;
- a korlátozó feltételek implicit függvényei a tervezési változóknak;
- a függvények, illetve a függvény gradiensek kiértékelése — a fenti okok miatt — rendkívül költséges.

A fenti nehézségek elkerülésére, illetve részleges kiküszöbölésére az optimalási feladat megfogalmazása, modellezése és annak megoldása során számos olyan egyszerűsítést és közelítést alkalmaznak (Kirsch (1993), Rozvany, Bendsoe és Kirsch (1995)), amely az így kapott eredményt, mint optimális megoldást kérdőjelezi meg.

2. A topológiai optimalási feladat

Vizsgálataink során feltételezzük, hogy a tartó lineárisan rugalmas anyag-moddal jellemezhető szerkezet. A tartó egyensúlyi állapota, a szerkezet geometriájának változása a rácsos tartó rúdjaiban keletkező feszültségek, valamint a csomóponti eltolódások függvényében meghatározhatók, illetve korlátozhatók. A topológiai optimalási feladat célfüggvénye lineáris, az egyensúlyi és kompatibilitási feltételek nemlineárisak és implicit függvényei a tervezési változóknak.

A geometriailag nemlineáris szerkezeti modellt a nagy elmozdulások elméletének alkalmazásával, a "Total-Lagrange" vonatkoztatási rendszerben ábrázoljuk. A deformált rácsos tartó egy rúdelemének megváltozott hossza:

$$l = \left[\sum_{d=1}^3 (l_d^0 + u_{2,d} - u_{1,d})^2 \right]^{\frac{1}{2}}. \quad (1)$$

ahol

l_d^0 a rúdelem deformálatlan hosszának a koordináta-tengelyekre vetített képe;

u_{1d} a rúdelem kezdőpontjának eltolódása;

w_{2d} a rúdelem végpontjának eltolódása.

A rácsos tartó teljes potenciális energia függvénye az alábbi formában adható meg:

$$V(u_i(a_j), a_j) = \frac{E}{2} \sum_{j=1}^e \frac{a_j}{l_j^0} (l_j^0 - l_j)^2 - p_i u_i(a_j) \quad (2)$$

$$i = 1, 2, \dots, n; \quad j = 1, 2, \dots, e$$

ahol

u_i a rácsos tartó csomóponti eltolódásait,

a_j a rúdelemek keresztmetszeti méreteit, mint tervezési változókat jelöli;

p_i a rácsos tartóra ható, külső, csomóponti teher vektora;

n a rácsos tartó csomóponti elmozdulásainak száma;

e a rácsos tartó rúdjaiknak száma;

E a szerkezet anyagának rugalmassági modulusa.

A térbeli rácsos tartó optimalizációs topológiáját az alábbi szélsőérték feladat megoldásával kapjuk. Határozzuk meg a térbeli rácsos tartó rúdelemeinek azon keresztmetszeti értékeit — megengedve a zérus keresztmetszeti értékeket is —, amelyek minimális súlyú szerkezetet eredményeznek.

$$w = \rho l_j^0 a_j \rightarrow \min, \quad (3)$$

kielégítik a

$$V_{,i}(u_i(a_j))|^{a_i} = 0, \quad (4)$$

egyensúlyi feltételeket és az

$$a_j \geq 0,$$

$$\underline{u} \leq u_i(a_j) \leq \bar{u}, \quad (5)$$

$$\underline{s} \leq s_j(u_i(a_j)) \leq \bar{s}, \quad (6)$$

$$i = 1, 2, \dots, n; \quad j = 1, 2, \dots, e,$$

egyenlőtlenségi korlátozó feltételeket, ahol

$\rho l_j^0 a_j$ célfüggvény a szerkezet súlya;

$V = V(u_i, a_j)$ a szerkezet teljes potenciális energia függvénye;

$V_{,i}|^{a_j} = 0$ a szerkezet egyensúlyi egyenletrendszere az adott a_j helyen;

\underline{u} a csomóponti elmozdulások alsó korlátja;

\bar{u} a csomóponti elmozdulások felső korlátja;

s_j a rácsos tartó rúdelemeiben keletkező feszültség;

\underline{s} a feszültségek alsó korlátja;

\bar{s} a feszültségek felső korlátja;

ρ az anyagsűrűség.

Az egyensúlyi egyenletrendszerre vonatkozó (4) kifejezésben $(\cdot)_{,i}$ az u_i csomóponti eltolódások szerinti parciális deriváltakat jelöli.

A rúdelemek alakváltozása és a rácsos tartó csomópontjainak elmozdulása közötti kompatibilitási feltétel a "Total-Lagrange" vonatkoztatási rendszer alkalmazása miatt minden esetben teljesül. A rúdelemekben keletkező feszültségek a lineárisan rugalmas anyagtörvény feltételezésével a deformált rúd hosszának ismeretében az

$$s_j = \frac{l_j^0 - l_j(u_i(a_j))}{l_j(u_i(a_j))} E \quad (7)$$

összefüggés alapján meghatározható.

3. A szerkezetoptimalási módszer

Az ismertetendő módszer egy aktív feltételeket kezelő külsőpontos eljárás, amely az optimális topológiát linearizált lépések sorozatával közelíti meg. Az egyes $r = 0, 1, 2, \dots$ lépéseket az $\{a_j^r, u_i^r, s_j^r\}$ vektorokkal jellemezzük. A Δa_j^r keresztmetszet növekményeket — mint irányvektort — minden lépésben egy-egy lineáris programozási feladat megoldásával nyerjük. A lineáris programozási feladatban a szerkezet súlynövekményét minimalizáljuk, a linearizált aktív elmozdulás és feszültség korlátok figyelembevételével.

A keresztmetszet növekmények $\alpha \Delta a_j^r$, $\alpha \leq 1$ nagyságát *iterative* a linearizálás hibájának korlátozásával, vagyis a lineárisan becsült és a tényleges lehajlások és feszültségek eltérésének korlátozásával állítjuk be. Az első lépésben α bemenő paraméter, ezután mindig az előző lépés eredményét használjuk induló α paraméterként.

A keresztmetszet növekmények meghatározása után az $a_j^r = a_j^{r-1} + \alpha \Delta a_j^r$ pontban Newton-eljárással megoldjuk a szerkezet nemlineáris egyensúlyi egyenletrendszerét, amely az adott pontban a szerkezet elmozdulásait, illetve feszültségeit adja. Az eljárást addig ismételjük, amíg el nem érjük a megvalósítható megoldások halmazát.

A módszer egy azonos ξ keresztmetszetű rúdelemekből álló $\{a_j^0, u_i^0, s_j^0\}$, $a_j^0 = \xi$, $\xi \approx 0$, nem megvalósítható alapszerkezetből indul ki. Az irányvektort adó lineáris programozási feladat felépítése a következő:

$$\rho l_j^0 \Delta a_j^r \rightarrow \min! \quad (8)$$

$$u_i^r + \widehat{\Delta} u_i^r \leq \bar{u}_i \quad \text{ha} \quad u_i^r \succ \bar{u}_i \quad (9)$$

$$u_i^r + \widehat{\Delta} u_i^r \geq \underline{u}_i \quad \text{ha} \quad u_i^r \prec \bar{u}_i \quad (10)$$

$$s_j^r + \widehat{\Delta} s_j^r \leq \bar{s}_j \quad \text{ha} \quad s_j^r \succ \bar{s}_j \quad (11)$$

$$s_j^r + \widehat{\Delta} s_j^r \leq \underline{s}_j \quad \text{ha} \quad s_j^r \prec \bar{s}_j \quad (12)$$

ahol

$\widehat{\Delta} u_i^r$ a csomóponti eltolódás növekmény közelítő értéke

$\widehat{\Delta} s_j^r$ a feszültség növekmény közelítő értéke.

A $\widehat{\Delta} u_i^r$ csomóponti eltolódás növekményt az alábbi lineáris approximációval határozhatjuk meg:

$$\widehat{\Delta} u_i^r = \alpha u_{i,j} \Delta a_j^r \quad (13)$$

$$i = 1, 2, \dots, n, \quad j = 1, 2, \dots, e,$$

ahol $u_{i,j}$ a rácsos tartó $V_{,k}(u_i) = 0$ egyensúlyi egyenletrendszeré alapján a következő módon számolható:

$$V_{,kj} + V_{,ki} u_{i,j} = 0, \quad (14)$$

$$u_{i,j} = [V_{,ki}]^{-1} V_{,kj}. \quad (15)$$

A $\widehat{\Delta} s_j^r$ feszültség növekmény közelítő értékének meghatározása az előzőekhez hasonló módon történik:

$$\widehat{\Delta} s_j^r = \alpha s_{j,l}^r \Delta a_l^r \quad (16)$$

$$j = l = 1, 2, \dots, e,$$

ahol:

$$s_{j,l}^r = s_{j,i}^r u_{i,l}, \quad (17)$$

mivel

$$\widehat{\Delta} s_j^r = s_{j,i}^r \Delta u_i^r, \quad (18)$$

amelyből a (13) kifejezés alapján a

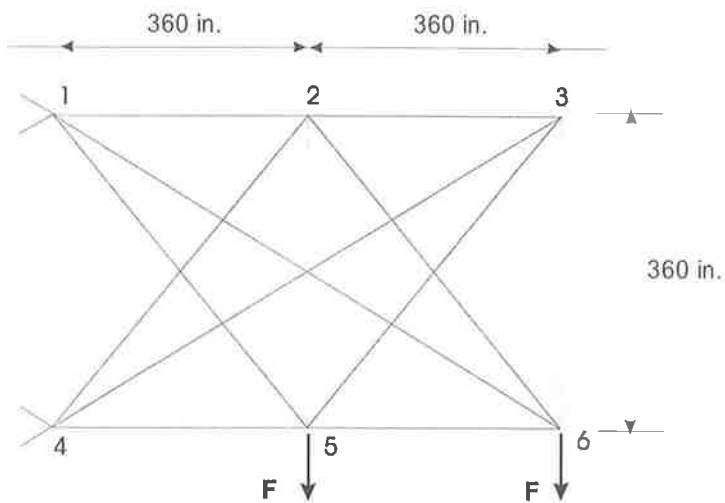
$$\widehat{\Delta} s_j^r = s_{j,i}^r u_{i,l}^r \Delta a_l^r, \quad (19)$$

összefüggést kapjuk.

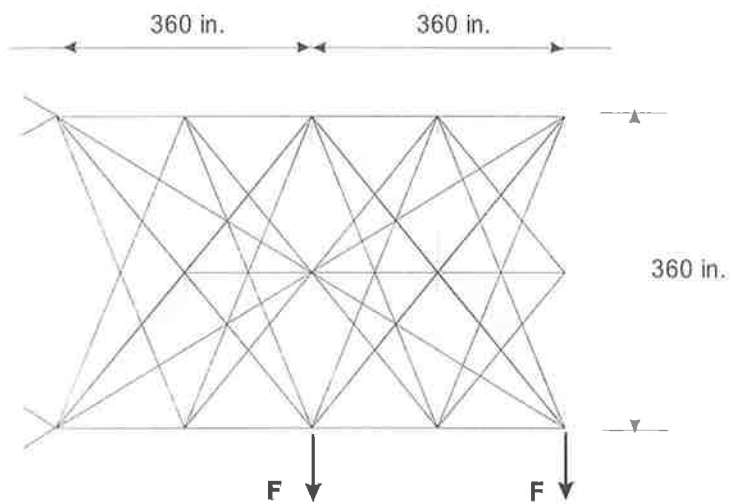
4. Numerikus példák

A módszer hatékonyságát és alkalmazási lehetőségeit az irodalomból jól ismert teszt feladatokon keresztül vizsgáltuk. A szerkezeti modell geometriailag nemlineáris, ideálisan rugalmas anyagú. A rugalmassági modulus 10^7 psi, anyagsűrűség $\rho = 0.1$ lb/in³. A szerkezetet az 1. és 2. ábrán megadott csomópontokban ható $F = 10^5$ lb nagyságú koncentrált erő terheli. A megfogási viszonyokat ugyancsak az 1. és 2. ábrák szemléltetik. A tervezés során azt a legkisebb súlyú szerkezetet kerestük, amely kielégíti az $u_{\max} = 2.0$ in lehajlás korlátot és az $s_{\max} = 25000$ psi feszültség korlátot.

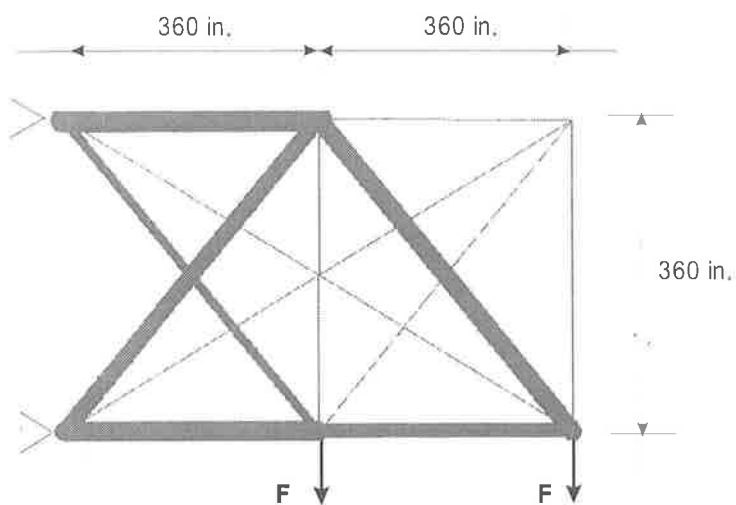
Az 1. ábra szerinti alapszerkezetből kiindulva a 3. ábrán látható eredményt kaptuk. A 2. ábrán egy olyan alapszerkezetet tüntettünk fel, amelyet az előző feladathoz képest "sűrűbb" pontrács felvételével nyertünk. Az így kapott eredmény az előzővel megegyező, melyet a 4. ábra szemléltet. Az optimális szerkezet súlya mindkét esetben 5036.67 lb és az eredményként kapott geometriai elrendezés is mindkét esetben megegyezik. A 3. és 4. ábrákon vékony vonalakkal ábrázolt rudak keresztmetszete a vastagabb vonalakkal jelölt rúdkeresztmetszetekhez képest olyan kicsi, hogy gyakorlatilag nullának tekinthető. Meg kell jegyeznünk azonban, hogy az eredmény minden esetben függ az alapszerkezet felvételétől és csak abban az esetben jutunk azonos eredményre, azaz csak abban az esetben kapjuk az optimális topológiai elrendezést, ha az eredmény részhalmaza az előre felvett alapszerkezetnek.



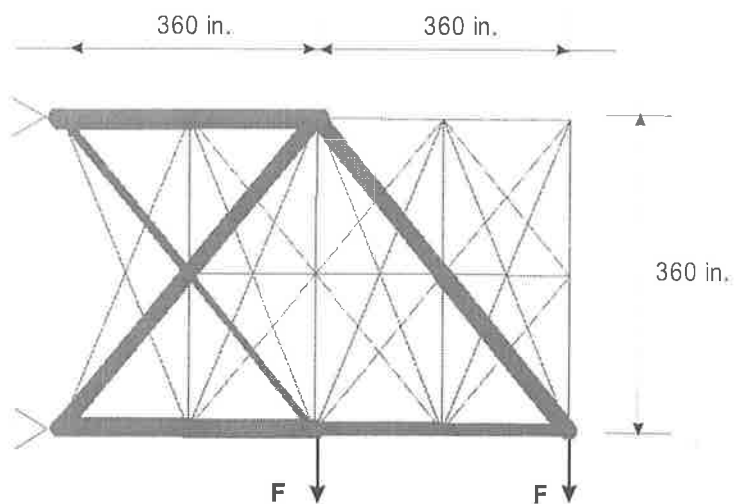
1. ábra



2. ábra



3. ábra



4. ábra

Irodalom

1. Ben-Tal A. és Bendsøe M. P. (1993), A new method for optimal truss topology design, *SIAM J. Optim.* 3, 323-358
2. Bendsøe M. P. és Kikuchi N. (1988), Generating Optimal Topologies in Structural Design Using a Homogenization Method, *Comp. Meth. Appl. Mech. Eng.* 71, 197-224
3. Forgó F. (1978), Nemkonvex és diszkrét programozás, Közgazdasági és Jogi Könyvkiadó, Budapest
4. Hemp, W. S. (1973), *Optimum Structures*, Clarendon Press, Oxford, UK.
5. Kirsch U. (1981), *Optimum structural design*, Mc Graw-Hill Book Company
6. Kirsch U. (1989), *Optimal Topologies of Structures*, *Applied Mechanics Reviews*, Vol. 42, 223-239
7. Michell, A. G. M. (1904), *The Limits of Economy of Material in Frame Structures*, *Philosophical Magazine*, Series 6, Vol. 8, 589-597
8. Rozvany GIN (1989), *Structural Design Via Optimality Criteria*, Kluwer, Dordrecht, The Netherlands
9. Rozvany GIN, Bendsøe M. P. és Kirsch U. (1995), *Layout optimization of structures*, *Applied Mechanics Reviews*, Vol. 48, 41-118
10. Suzuki K. és Kikuchi N. (1991), A Homogenization Method for Shape and Topology Optimization, *Comp. Meth. Appl. Mech. Eng.* 93, 291-318

A NEW EXTERIOR POINT METHOD FOR OPTIMAL TRUSS TOPOLOGY DESIGN

In this paper an exterior point method is presented for optimal truss topology design using a non-linear structural model. The trusses — pin-jointed frames — have many practical applications in building construction and bridge as well as in aerospace engineering. The objective of optimization is the weight of the structure. The behavior constraints for member stresses and nodal displacements are implicit functions of design variables. Truss topology optimization based on the assumption of an initial ground structure and formulated in terms of member cross-sectional areas, member stresses and nodal displacements results a large, non-linear and non-convex optimization problem. The method proposed is computationally attractive and have been tested on large number of applications, some of which are presented.

KÖNYVEKRŐL

DANYI PÁL – VARRÓ ZOLTÁN: *Operációkutatás üzleti döntések megalapozásához*. Pécs, 1997. JPTE Kiadó, 411 o.

Danyi Pál és Varró Zoltán több évtizedes oktatói és kutatói tapasztalatokkal a hátuk mögött vállalkoztak egy új szemléletű, az operációkutatás fejlődésének számos fontos területét bemutató egyetemi tankönyv megírására.

A tankönyv megírását több körülmény is szükségessé tette. Az egyik, talán legfontosabb elem, a karunkon több éve folyó tantárgy-korszerűsítési és képzés-fejlesztési program, melynek eredményeképpen lényeges tartalmi és szerkezeti változások következtek be a Gazdaságmatematikai Tanszék által oktatott tárgyak tekintetében. Új felfogásban és megváltozott tematikával oktatjuk a 90-es évek eleje óta az alapozó matematikai tárgyakat, illetve az operációkutatást. A koncepcionális változások miatt a korábban használt jegyzeteink már nem feleltek meg képzési céljainknak, elavultak. Miután a hazai piacon nem létezett képzési elképzeléseinkhez illeszkedő Operációkutatás könyv, ezért nem volt más választás, mint ennek a könyvnek a megírása.

A tárgyak egymásra épülésének a logikájából következett, hogy először az alapozó matematikai tárgyak szükségleteit kielégítő tankönyvek készültek el. Az optimalizáláselméletnek ugyanis számos olyan területét, amely az operációkutatás megalapozásához elengedhetetlenül szükséges, az általam írt "Bevezetés egysúlyi és optimalizáló modellek vizsgálatának matematikai módszereibe" című tankönyv részletesen tárgyalja. Ez lehetővé tette bizonyos ismétlések elkerülését, viszont igényelte a hivatkozások, utalások pontos alkalmazását (fejezet, tétel, esetleg oldalszám stb.)

A tankönyv nemcsak a bevezető matematikai tárgyra épít nagyon didaktikusan, hanem számos további tárgy számára is ad kapcsolódási lehetőségeket. Különösen érvényes ez a Termelésmenedzsment című tárgyra. A tankönyv számos fejezete jól megalapozza a termelés-menedzsment egy-egy fontos témakörét.

Ez a könyv folytatása a szerzők 1995-ben megjelent "Lineáris programozás" című tankönyvének. A szerzők szándéka szerint a könyv nem kíván monográfia lenni, alapvetően és elsősorban az egyetemi képzésünk igényeinek akar hosszabb távon is megfelelni. Ebből eredően ez a könyv nem tartalmazza az összes olyan témakört, amelyek manapság az operációkutatáshoz tartoznak. Nem szerepelnek benne például a sztochasztikus problémák (sorbanállások elmélete, készletgazdálkodás, Markov-láncok stb.). Ezt a hiányt részben Vörös József "Termelésmenedzsment" című könyve pótolja képzésünkben.

A könyv elsősorban közgazdász egyetemi hallgatók számára készült, azonban számos olyan modell és megoldási módszert is tárgyal, amely posztgraduális kurzus tananyagát képezheti. Ennek megfelelően alapos ismereteket tételez fel lineáris algebrából, lineáris programozásból és többváltozós analízisből.

A könyv hét, meglehetősen különböző hosszúságú fejezetből áll. Ezek nem tükrözik vissza a bemutatott modellek és módszerek fontosságát, inkább a szerzők egyéni ízlése alakította ki egy-egy fejezet terjedelmét.

Az első (kissé eklektikusra sikerült) fejezet a címe szerint lineáris feltételrendszerű matematikai programozási modellek megoldási módszereit mutatja be. Az euklideszi terekkel kapcsolatos alapfogalmak után a konvex (konkáv) függvények fogalmát általánosítja és bevezeti a kvázikonvexitást. A szétválasztható változójú programozási feladatok besorolása e fejezetbe kissé esetlegesnek tűnik, hiszen a bemutatott módszer a konvex, tehát általánosabb feltételi függvények esetét tárgyalja. Az utolsó alfejezet az LP feladatok megoldása terén elért legújabb eredményeket foglalja össze, majd egy affin belső pontos algoritmussal és Karmarkar projektív módszerének egyszerűsített változatával is megold egy-egy LP feladatot.

A második fejezet a többcélú optimalizálás módszereit és a célprogramozást ismerteti meg az olvasóval. A fejezet végén több rövidebb-hosszabb gyakorlati probléma segít a célprogramozási modellek készítésének elsajátításában.

A hálózati modelleket tárgyaló harmadik fejezet a minimális élsúlyú feszítőfa problémával, továbbá a minimális költségű hálózati folyam problémával és annak speciális eseteivel (átrakási, szállítási, hozzárendelési, legrövidebb út és maximális folyam probléma) foglalkozik. A szállítási és hozzárendelési feladat megoldása a korábbi kötetben található, azonban az utolsó alfejezet rávilágít a magyar módszer hátterére.

A negyedik fejezet az egészértékű programozás legismertebb modelljeit és módszereit tárgyalja. Az elején bemutatja, hogy milyen módon lehet egész értékű változók segítségével az LP modelleket a valóságot hűbben tükröző modellekké alakítani. Felhívja a figyelmet, hogy gyakran nem csupán egyetlen helyes modell készíthető és példákat mutat arra, hogy milyen elvek alapján készíthetők viszonylag kis erőfeszítéssel megoldható modellek. Részletesen tárgyalja korlátozás és szétválasztás módszerét. Példákon keresztül mutatja be a legjobb korlát és a LIFO felbontási stratégiákat, sőt a szokásos LP lazítás mellett az utazó ügynök problémát hozzárendelési lazítással is megoldja. Számos speciális 0-1 értékű modell (halmazlefedési, halmazfelbontási, halmazkitöltési és kvadratikusan hozzárendelési feladat), továbbá hálózati modellek (párosítási, súlyozott párosítási és kínai postás probléma) találhatók a fejezet hátralévő részében.

A dinamikus programozást bemutató ötödik fejezet számos korábról ismert probléma ismételt megoldásával mutatja be a módszer alap gondolatát. Ismét nagy súlyt fektet a különböző modellek és módszerek kapcsolatára. A hátizsák feladatra kétféle hálózati modellt is megfogalmaz, továbbá feladatként tűzi ki az utazó ügynök és az LP feladat megoldását dinamikus programozással.

Külön alfejezetet szentel a dinamikus sorozatnagyság problémának. A fix költséges modellre három megoldási lehetőséget is mutat: egy a feladat speciális struktúráját ki nem használó egész értékű modellt, Wagner és Whitin jól ismert dinamikus programozási algoritmusát, továbbá két olyan tanulmányos egész értékű modellt, amelyeknek elegendő az LP lazítását megoldani.

A játékelméletről szóló fejezet csak a mátrixjátékok LP megoldását és a folytonos játékok közül az oligopol játékot tárgyalja. Az üzleti életben oly fontos nem zérus összegű játékok kihagyása mindenképpen hiányérzetet kelthet az olvasóban.

Az utolsó fejezet a bonyolultságelmélet alapjaiba enged rövid betekintést, majd néhány ismert probléma megoldásának legrosszabb eset elemzését végzi el. E röpké fejezet fontossága abban rejlik, hogy a modellalkotónak ismerni kell a modell megoldásához szükséges erőfeszítéseket. A legrosszabb eset elemzés alapján alkotott kép ugyan gyakran félrevezető a gyakorlati megoldhatóság szempontjából, arra azonban mindenképpen alkalmas, hogy az olvasó mintegy összefoglalásként ismét áttekinthesse az egyes algoritmusok főbb lépéseit.

Véleményem szerint a Danyi-Varró szerzőpár nagyon igényes, magas színvonalú tankönyvet írt. A feldolgozott témakörök az operációkutatás legfontosabb területeit mutatják be. A tárgyalásmód világos, a szerzők nagy tapasztalattal döntenek arról, hogy mikor közöljék egy-egy tétel precíz matematikai bizonyítását és mikor lehet megelégedni egy-egy szemléletes magyarázattal, vagy egyszerűen az állítás megfogalmazásával. Mivel nem matematikusok részére készült a tankönyv, ez a "lazaság" inkább előnye a kifejtésnek, mintsem hiányossága.

Mivel a szerzők fontosnak tartják, hogy a megcélzott olvasóközönség a gazdasági és üzleti problémákat képes legyen matematikai modellekkel megfogalmazni, ezért súlyt fektettek a modellezési készség fejlesztésére. A gyakorlat nagyméretű modelljeinek hatékony megoldásához azonban szükség van a megoldási módszerek ismeretére is, hiszen ez nyújt támpontot az alkalmas modell és algoritmus vagy számítógépes programcsomag kiválasztásában. A szükséges adatbázis összeállításánál és az eredmények értelmezésénél is gyakran fontos a megoldási módszer ismerete.

A tankönyv további érdeme, hogy nagyon sok feladatot közöl, melyek nagy része "valóságos" feladat a business, marketing és menedzsment területéről.

A tankönyv technikai kivitelezése színvonalas. Sok ábra, diagramm, rajz magyarázza az egyébként nem túl egyszerű algoritmusokat, könnyítve ezáltal a gondolati feldolgozást.

Külön is kiemelendő, hogy a szerzők minden fejezet végén egy rövid történeti áttekintést adnak az éppen megtárgyalt témakörrel, valamint jelentős bibliográfiát is közölnek, mely elsősorban angol nyelvű, de a hazai szerzők munkássága is megfelelő módon reprezentálva van.

Véleményem szerint a tankönyv országos vonatkozásban is hiánypótló. Számos olyan új eredményt, módszert is tárgyal, amely a közelmúltban született és magyar nyelvű szakkönyvek még nem dolgozták fel azokat.

A tankönyv tudatosan úgy íródott, hogy különböző képzési szinteken és formákban tanulóknak is tananyagul szolgálhat. Bizonyos fejezetei kifejezetten a Gazdálkodástani doktori képzésünk számára készült. Meggyőződésem, hogy akár kötelező, akár ajánlott irodalomként más felsőfokú intézményben is jól hasznosulhat ez a munka.

Komlói Sándor

Janus Pannonius Tudományegyetem
KTK Gazdaságmatematikai Tanszék